Near-Optimal Thermal Monitoring Framework for Many-Core Systems-on-Chip

Juri Ranieri, *Member, IEEE*, Alessandro Vincenzi, *Member, IEEE*, Amina Chebira, *Member, IEEE*, David Atienza, *Senior Member, IEEE*, and Martin Vetterli, *Fellow, IEEE*

Abstract—Chip designers place on-chip thermal sensors to measure local temperatures, thus preventing thermal runaway situations in many-core processing architectures. However, the quality of the thermal reconstruction is directly dependent on the number of placed sensors, which should be minimized, while guaranteeing full detection of all the worst case temperature gradient. In this paper, we present an entire framework for the thermal management of complex many-core architectures, such that we can precisely recover the thermal distribution from a minimal number of sensors. The proposed sensor placement algorithm is guaranteed to reduce the impact of noisy measurements on the reconstructed thermal distribution. We achieve significant improvements compared to the state of the art, in terms of both computational complexity and reconstruction precision. For example, if we consider a 64 cores systems-on-chip with 64 noisy sensors ($\sigma^2 = 4$), we achieve an average reconstruction error of 1.5°C, that is less than half of what previous state-of-the-art methods achieve. We also study the practical limits of the proposed method and show that we do not need realistic workloads to learn the model and efficiently place the sensors. In fact, we show that the reconstruction error is not significantly increased if we randomly generate the power-traces of the components or if we have just a part of the correct workload.

Index Terms—Sensor placement, thermal management, thermal monitoring

1 Introduction

Technological advancements of the lithographic process steadily increase the amount of components that can be placed on a single die. If we assume that the power consumed by these components does not decreases significantly with the technological progress [1], we have an increase of the power density, and subsequently, of the produced heat.

Many-core system-on-chip (SoC) have their performance limited by such increased heat density. More precisely, unfavorable thermal patterns increase the overall failure rate of the system [2], reduce performance [3], significantly increase leakage power consumption and cooling costs [2], [4].

In the past, passive thermal management schemes were used to limit the problems induced by thermal phenomena. For example, designers would organize the floorplan by placing the highest power density components closer to the heat sink [5]. However, in recent architectures such components are not easily identifiable since they depend on the workload execution patterns and, unfortunately, these patterns are not fully known at design time. Furthermore, these issues are amplified in many-core designs, where thermal hot-spots are generated without a clear spatio-temporal pattern due to the dynamic task set execution nature, based on external service requests, as well as the dynamic assignment to cores by the many-core OS [4], [6]. An example of these architecture and their critical thermal behavior is shown in Fig. 1, where you can observe the layout of a 64-cores architecture designed by STM [7], [8] and an example of its thermal distribution at run-time. Note how the cores are not anymore regularly organized and how they tend to spread non-uniformly. Such irregularity is forced by the complex constraints imposed during the floorplanning optimization and generates irregular thermal distributions with possibly many unpredictable hot-spots.

Thermal sensors have been already included into SoC designs [7], however their position have only been manually tuned because the knowledge of the temperature in a couple of locations was, until now, sufficient. Nowadays, it has become necessary to precisely measure the temperature distribution of the entire die and optimize the workload of the different components to maximize the performance while avoiding hotspots or large gradients of temperature.

At the same time, each temperature sensor has a significant impact in terms of occupied area and consumed power, therefore we would like to place as few sensors as possible. It is not yet clear how to optimize their placement to maximize the collected information about the thermal distribution. In fact, such aspects are quite complex and recently received significant attention [9], [10], [11], [12], [13], [14], [15].

In this paper, we consider the following two subproblems as core to design an efficient thermal monitoring system:

- *Thermal distribution reconstruction*. Given the temperature measured with L sensors at known locations,

- *J. Ranieri and M. Vetterli are with the School of Computer and Communication Sciences, Ecole Polytechnique Fédérale de Lausanne (EPFL), CH-1015 Lausanne, Switzerland. E-mail: {juri.ranieri, martin.vetterli}@epfl.ch.*
- *A. Vincenzi and D. Atienza are with the School of Engineering, Ecole Polytechnique Fédérale de Lausanne (EPFL), CH-1015 Lausanne, Switzerland. E-mail: {alessandro.vincenzi, david.atienza}@epfl.ch.*
- *A. Chebira is with the Centre Suisse d'Electronique et Microtechnique (CSEM), CH-2002 Neuchtel, Switzerland. E-mail: amina.chebira@csem.ch.*

Manuscript received 18 June 2014; accepted 23 Dec. 2014. Date of publication 21 Jan. 2015; date of current version 9 Oct. 2015.
Recommended for acceptance by H. Sarbazi-Azad.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TC.2015.2395423

how do we precisely estimate the temperature distribution of the whole die?

- *Sensor placement optimization*. Given a fixed amount of sensors, where do we place them so that we maximize the precision of the reconstructed thermal distributions?

Then, we propose to base the solutions of the aforementioned two sub-problems on the use of a linear low-dimensional subspace to represent the thermal distributions. Such models are interesting because they are sufficiently precise while being extremely simple, thus requiring limited computational resources.

Linear models have been already considered in the past for thermal monitoring applications [9], [14], but many questions have been raised regarding their practical feasibility.

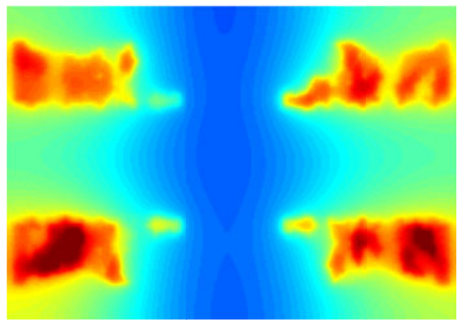
First, such models must be optimized and they require a set of thermal distributions representing the operations of the SoC under all the possible workloads. It is clear that such data is hard to gather at the design phase; that is when we would like to optimize the sensors locations. More precisely, we need to know three main inputs to simulate the thermal distributions: the floorplan, the workload of the SoC, and the power traces of the components under such workload. In this paper, we show that it is not necessary to have an exact description of the typical workload at design-time to reconstruct precisely the thermal distribution at run-time. In fact, we show that it is possible to optimize the model and place the sensors using a randomly generated workload without having significant losses in terms of reconstruction error.

Second, the performance of linear models depends strongly on the locations of the sensors. In fact, the noise corrupting the collected measurements can be dramatically amplified if the sensors are misplaced. We propose to optimize the sensor placement using FrameSense, a greedy algorithm based on a theoretical framework that we recently proposed [16]. Under some mild assumptions regarding the linear model, we can guarantee that the proposed algorithm is near-optimal in terms of reconstruction error and increases the noise stability of the framework. Moreover, such theoretical framework has already shown appealing performance in other domains, such as adaptive sensor scheduling [17].

The proposed framework improves significantly the monitoring performance over the state of the art. For example, consider the 64-cores architecture shown in Fig. 1a and typical scenario of one temperature sensor per core corrupted by a noise with variance $\sigma^2 = 4$, we can improve the reconstruction error from $\approx 3^\circ\mathrm{C}$ to less than $1.4^\circ\mathrm{C}$. Even if we increase the amount of temperature sensors by 50 percent, that is from 64 to 96, the previous methods [11], [15] can only reach an average reconstruction error of $\approx 2.4^\circ\mathrm{C}$, while our method can go down to almost $1^\circ\mathrm{C}$.

The remainder of the paper is organized as follows: in Section 2 we describe previous approaches and state-of-the-art methods for the recovery of the thermal distribution of SoC designs. We state and provide a solution to the problem of thermal reconstruction and the relative sensor placement in Section 3.1. We conclude with a thorough z comparative analysis of the performance of the proposed methods in Section 4 by means of extensive numerical simulations.

(a)



(b)

Fig. 1. (a) The layout of the considered 64 cores processor designed by STM, where each color represents a different core connected to its own L1 cache. The white part contains components with a limited thermal impact. (b) An example of a thermal distribution of such a processor, where the colormap spans the temperature between $65^\circ\mathrm{C}$ (dark blue) and $90^\circ\mathrm{C}$ (red).

2 Background and Related Work

The thermal distribution of a SoC can be estimated using three different strategies:

- Solution of the direct problem, given the heat sources and the physical model of the temperature diffusion process.
- Solution of an optimization problem, given the value of the temperature in some locations and some a-priori model for the thermal distributions.
- Empirical approaches, where the thermal distribution is estimated by means of external devices, such as infrared cameras.

The first approach requires the knowledge of the heat sources, that can be ascribed to the knowledge of the detailed power consumption of the different components. Often, performance counters [18], [19] are used to estimate the power traces at run-time. However, the estimation of the thermal distributions from the power traces is a computationally expensive task, requiring the complex thermal models characterizing the thermal dissipation of the SoC. Recently, [20] proposed to reduce the complexity of these methods by using directly the performance counter to estimate the temperature, without the intermediate step represented by the power traces.

On the other hand, the optimization problems are generally ill-posed. In fact, it is impossible to solve the inverse problem from few, spatially localized, noisy measurements without some a-priori constraints on the thermal map, such as limited bandwidth [9]. The performance is significantly impacted by the small number of available sensors and the

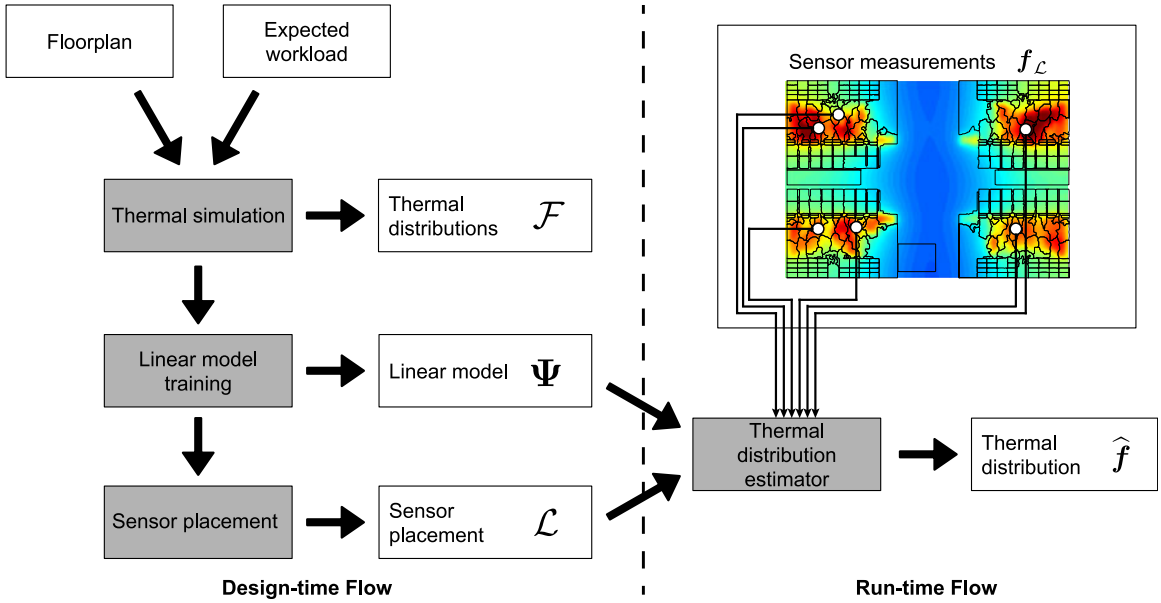


Fig. 2. Flow of the proposed framework, where data structures and algorithms are depicted with white and gray blocks, respectively. Note that the flow is divided in two parts: one for the design of the system and one for the run-time operations. The inputs to the systems are the floorplan of the SoC and the expected workload, while the outputs are a training set $\mathcal{F}$ of thermal distributions f, the linear model Ψ obtained from the training set $\mathcal{F}$, the sensor placement $\mathcal{L}$ and the estimated thermal distribution $\hat{f}$.

structure we consider for the thermal map, i.e. the a-priori information. Nowroz et al. [11] proposed a low-pass approximation strategy to reduce the number of sensors that are placed using an *energy-based* algorithm. This sensor allocation algorithm has been improved by Reda et al. [12] using a heuristic iterative approach to approximate an NP-hard problem. The authors in [21] proposed a grid-based uniform sensor placement followed by interpolation to approximate the temperature. These works estimated entire thermal maps, but the precision of the estimates is limited by the sub- optimality of the chosen models for the thermal distribution.

Other works have notable performance but are not focused on the estimation of the entire thermal map. Namely, the approach in [22] employs the correlation in power distribution to estimate the expected value of temperature at different locations of the chip using a dynamically tuned Kalman filter. The problem of noisy measurements has also been already considered; for example, a method based on the correlation between the different sensor has been presented in [23].

Recently, different researchers studied the estimation of the entire thermal map based on the temperature correlation between different locations [14], [15]. First, we have proposed in [14] an approach where we approximate the data with a low dimensional linear model based on such correlations. Such method brought an intuitive interpretation of the sensor placement problem together with appealing performance. However, we have showed in [16] that the sensor placement algorithms can be further improved. Contemporaneously, Zhou et al. proposed a reconstruction algorithm and an optimization of the sensor placement based on information theory [15]. Their reconstruction algorithm is substantially equivalent to the one proposed in [14], without the low-dimensional approximation. Such difference, as we will see in the numerical experiments, has a significant impact on the stability w.r.t. the noise affecting the measurements.

The methods based on external cameras [12], [24], [25], [26], [27] are generally considered to be the most precise ones. However, the increased precision comes at the cost of practical considerations. In fact, such methods cannot be used for run-time operations and are generally studied for two different purposes: the calibration of the on-chip temperature sensors [27] or the study of the thermal behavior of prototypes at design phase [12], [24], [25], [26].

We underline the existence of hybrid methods, mixing techniques taken by the different methods. For example, the authors of [13], [22] propose to use the information coming from the thermal sensors together with the performance counters to estimate the thermal distribution. This approach reduces the computational complexity of the methods solely based on performance counters and mitigates the effects of the noise corrupting the thermal sensors. The fusion and the integration of the two data sources to obtain the thermal distributions is usually accomplished by a Kalman filter. Such implementations have shown the ability to track precisely the temperature profile at the cost of the computational complexity, which is significantly higher than standard approaches.

3 A Near-Optimal Thermal Monitoring Framework

In this paper, we propose a framework for the problem of thermal monitoring of a many-core SoC. We assume to know the floorplan of the SoC and the time-varying power consumption of the components when handling some expected workload. The proposed framework is divided in two parts: design-time and run-time algorithms. We give a visual description of the framework in Fig. 2. At design-time, we have three main phases:

- Thermal simulation, where we generate a set $\mathcal{F}$ of thermal distributions representing the thermal behavior of the considered SoC.
- Model learning, where we learn the structure of the thermal distribution from the given set $\mathcal{F}$. We call the model $\boldsymbol{\Psi}$ and it has a fundamental role both for the reconstruction of thermal distribution and the optimization of the sensor placement.
- Sensor placement optimization, where we choose the optimal sensor positions $\mathcal{L}$ according to the model $\boldsymbol{\Psi}$.

The thermal simulation part is not covered in this paper, since it is possible to choose different algorithms and methods depending on the SoC architecture. In our case, we use 3D-ICE [28], a flexible fast compact transient thermal model for the thermal simulations of SoCs. This thermal simulator consider both the dynamic and static power consumptions. The static power is modeled in the system as a additive percentage of the dynamic power, that is mostly consumed in those units that are not active.

At run-time, the system is extremely simple: we collect the measurements from the sensors and, knowing the linear model and the sensors positions, we estimate the thermal distribution using an optimal least square estimator. Note that we propose the algorithm for the reconstruction but we do not study the details of its implementation, that should be adjusted according to each specific architecture.

In what follows, we describe each part of the framework in terms of mathematical abstraction and algorithmic solutions.

3.1 Sensing and Recovery of Thermal Distributions

We start the description of the proposed framework from the run-time phase. More precisely, we mathematically state the concept of temperature sensing and the recovery of the thermal distribution using the sensed data.

Consider a given SoC die and its thermal distribution $f(\boldsymbol{x})$ at a given time, where $\boldsymbol{x}$ indicates the multi-dimensional spatial location. While $f(\boldsymbol{x})$ is a continuous spatial function, we consider it to be discretized and vectorized as a vector $\boldsymbol{f} \in \mathbb{R}^N$, where N represents the desired resolution. Note that the vectorization of the thermal distribution does not induce any loss of information.

We denote the set indicating the $L < N$ sensor positions as $\mathcal{L}$ and we define the measured temperatures as $\boldsymbol{f}_{\mathcal{L}} \in \mathbb{R}^L$. Note that the subscript $\mathcal{L}$ indicates that we kept only the elements of $\boldsymbol{f}$ indexed by the elements of $\mathcal{L}$.

The problem of recovering a complex information, such as the thermal distribution $\boldsymbol{f}$, from a limited number of measurements $\boldsymbol{f}_{\mathcal{L}}$, must rely on some hidden structure available in the thermal distribution. In fact, without a structure we would face an undetermined problem and it would be impossible to uniquely recover the data $\boldsymbol{f}$ from the measurements $\boldsymbol{f}_{\mathcal{L}}$.

While there exists many strategies to model data structures, the thermal monitoring scenario constraints the choice to models allowing a fast, efficient and reliable reconstruction. In this work, we consider a linear subspace model, such as the one in [14]. More precisely, given a K-dimensional linear model defined by a matrix $\boldsymbol{\Psi} \in \mathbb{R}^{N \times K}$, we model the temperature distributions as,

$$\boldsymbol{f} = \boldsymbol{\Psi}\boldsymbol{\alpha}, \tag{1}$$

where $\boldsymbol{\alpha} \in \mathbb{R}^K$ is the K-dimensional parametrization of $\boldsymbol{\Psi}$. In other words, if the model $\boldsymbol{\Psi}$ is sufficiently precise for the thermal distributions, then $\boldsymbol{\alpha}$ is a compact representation of $\boldsymbol{f}$. Namely, knowing $\boldsymbol{\alpha}$ is equivalent to know the temperature distribution $\boldsymbol{f}$.

When we consider a set of measurements $\boldsymbol{f}_{\mathcal{L}}$ we obtain the following pruned linear system,

$$\boldsymbol{f}_{\mathcal{L}} = \boldsymbol{\Psi}_{\mathcal{L}}\boldsymbol{\alpha}, \tag{2}$$

where $\boldsymbol{\Psi}_{\mathcal{L}} \in \mathbb{R}^{L \times K}$ is the collection of L rows of $\boldsymbol{\Psi}$ indexed by $\mathcal{L}$. We note that if $L \leq K$ and if $\operatorname{rank}(\boldsymbol{\Psi}_{\mathcal{L}}) = K$, we can uniquely reconstruct $\boldsymbol{f}$ from the measured data $\boldsymbol{f}_{\mathcal{L}}$ as

$$\widehat{\boldsymbol{f}} = \boldsymbol{\Psi}{\boldsymbol{\Psi}_{\mathcal{L}}}^{+}(\boldsymbol{f}_{\mathcal{L}} + \boldsymbol{\omega}), \tag{3}$$

where $\widehat{\boldsymbol{f}}$ is the estimated thermal distribution, $\boldsymbol{\omega}$ represents the noise in the measurements and ${\boldsymbol{\Psi}_{\mathcal{L}}}^{+}$ is the Moore-Penrose pseudoinverse, defined for a generic matrix $\boldsymbol{A}$ as

$$\boldsymbol{A}^{+} = (\boldsymbol{A}^{*}\boldsymbol{A})^{-1}\boldsymbol{A}^{*},$$

where $\boldsymbol{A}^{*}$ is the conjugate-transpose of $\boldsymbol{A}$. Note that we are using a least-square estimator that minimizes the large deviations, such as hot-spots.

Algorithm 1. Thermal Distribution Estimator

Require: Linear model $\boldsymbol{\Psi}$, Sensor locations $\mathcal{L}$, thermal distribution mean μ.
Ensure: Estimated thermal distribution $\widehat{\boldsymbol{f}}$.
1: Collect the sensor measurements $\boldsymbol{f}_{\mathcal{L}}$.
2: Estimate thermal distribution: $\widehat{\boldsymbol{f}} = \boldsymbol{\Psi}{\boldsymbol{\Psi}_{\mathcal{L}}}^{+}(\boldsymbol{f}_{\mathcal{L}} - \mu_{\mathcal{L}}) + \mu$.

The pseudo code of the thermal distribution estimator is given in Algorithm 1, where we note the extreme simplicity of the operations.

Such an estimator is optimal if the noise $\boldsymbol{\omega}$ satisfies certain statistical conditions, such as being i.i.d. Gaussian. The performance of such an estimator depends mostly on two aspects: the quality of the linear model and the optimization of the sensor locations $\mathcal{L}$.

More precisely, the following two questions must be answered to validate and strengthen the proposed framework:

- How can we design a reliable and precise model $\boldsymbol{\Psi}$?
- How do we choose the optimal sensor placement $\mathcal{L}$?

In what follows, we discuss and answer these two questions.

3.2 Training the Linear Model for Thermal Distributions

Assume that we are given a representative set of M possible thermal maps $\mathcal{F} = \{\boldsymbol{f}_i\}_{i=1}^{M}$ and we would like to find a model $\boldsymbol{\Psi}$ that can precisely represent such a dataset with K-dimensional parameter $\boldsymbol{\alpha}$. In theory, the set $\mathcal{F}$ should represent all the possible thermal distributions that the SoC could produce during operations. While this assumption is necessary in theory, we will show in the numerical experiments that we can relax it significantly.

Given the model $\boldsymbol{\Psi} \in \mathbb{R}^{N \times K}$ and a thermal distribution $\boldsymbol{f}$, we obtain the approximated thermal distribution $\widetilde{f}$ by the following projection onto the model,

$$\widetilde{\boldsymbol{f}} = \boldsymbol{\Psi}\boldsymbol{\Psi}^{+}\boldsymbol{f}. \tag{4}$$

Then, we evaluate the quality of the model by measuring the the approximation error ϵ, that is defined as

$$\epsilon = \mathsf{E}_{\mathcal{F}}\|\boldsymbol{f} - \widetilde{\boldsymbol{f}}\|_2, \tag{5}$$

where the subscript $\mathcal{F}$ indicates that the expectation is taken over all the training set $\mathcal{F}$.

In what follows, we propose to learn the model $\boldsymbol{\Psi}$ from the principal component analysis (PCA) of the temperature distributions. Note that we considered also other models, such as the one based on non-negative matrix factorization, but their performance were not sufficiently interesting to be included in this paper.

3.2.1 Principal Component Analysis

Given $\mathcal{F}$, the PCA generates the model $\boldsymbol{\Psi}$ formed by a set of K *orthonormal* vectors, known as principal components. The PCA is defined so that the first principal component corresponds to the direction of the largest variance (i.e. representing as much of the variability in the data as possible), and each successive component has the highest variance in the subspace orthogonal to one spanned by the preceding components. Therefore, the PCA finds an orthonormal basis that spans the K dimensional subspace containing the largest amount of information on $\mathcal{F}$ (in the MSE sense). In other words, the PCA generates the optimal K-dimensional subspace that minimizes the approximation error of the given training set $\mathcal{F}$.

The solution of the PCA can be analytically computed as the K eigenvectors corresponding to the K largest eigenvalues of the correlation matrix $\boldsymbol{\Sigma}_{\boldsymbol{f}} = \mathsf{E}[\boldsymbol{f}\boldsymbol{f}^{*}]$. Consequently, most of the technical difficulties of PCA relate to the estimation of $\boldsymbol{\Sigma}_{\boldsymbol{f}}$, in particular when the data is incomplete or noisy.

PCA has been already proposed to model thermal distributions and showed promising results [14]. In fact, if the thermal distributions of $\mathcal{F}$ are well approximated by a K dimensional subspace, then the PCA generates the optimal model $\boldsymbol{\Psi}$.

In certain scenarios, characterized by a limited amount of available resources, we may prefer other methods. For example, certain architectures cannot afford the memory load to store the matrix $\boldsymbol{\Psi}$. Prior works [11], [9] proposed to use models based on the discrete cosine transform (DCT). Such models have a clear advantage in terms of memory used, since they do not require to store $\boldsymbol{\Psi}$ in the system.

Unfortunately, the DCT based models do not outperform the PCA model as we do not design entirely $\boldsymbol{\Psi}$. Moreover, it is not possible to exploit all the traditional computational advantages of DCT transforms. In fact, while we can compute efficiently the DCT transform, once we select some rows to represent the sensors measurements we destroy the structure of the DCT transform and loose any computational advantage.

A comparison of the reconstruction performance between PCA and DCT is proposed in Section 4 and it shows that PCA is always the best choice unless there exist stringent limits on the memory available for runtime operations. Therefore, we assume to use the PCA for the *linear model training* and we show its pseudo-code in Algorithm 2.

Algorithm 2. Linear Model Training

Require: Training set $\mathcal{F}$.
Ensure: Linear model $\boldsymbol{\Psi}$, thermal distribution mean μ.
1: Compute the mean: $\mu = \mathsf{E}_{\mathcal{F}}[\boldsymbol{f}]$.
2: Compute the covariance matrix: $\boldsymbol{\Sigma}_{\boldsymbol{f}} = \mathsf{E}\mathcal{F}[\boldsymbol{f}\boldsymbol{f}^{*}]$.
3: Compute the first K eigenvectors of $\boldsymbol{\Sigma}_{\boldsymbol{f}}$.
4: Form $\boldsymbol{\Psi}$ by using the eigenvectors as its columns.

3.3 Optimization of the Sensor Placement

As explained in Section 3.3, the reconstruction of thermal distributions relates to precisely estimating $\widehat{\boldsymbol{f}}$ from possibly noisy measurements $\boldsymbol{f}_{\mathcal{L}}$.

In a typical scenario, we are given a number of sensors L and a set $\mathcal{P}$ of P possible locations, that is a subset of the area of the SoC. Moreover, assume that either we have or can find an optimized linear model $\boldsymbol{\Psi}$ for the thermal distributions $\mathcal{F}$, as described in Section 3.2. Then, we would like to find the sensor placement $\mathcal{L}$ that minimizes the reconstruction error ξ of the thermal distributions,

$$\xi = \mathsf{E}\|\boldsymbol{f} - \widehat{\boldsymbol{f}}\|_2^2. \tag{6}$$

The reconstruction error depends mostly on the eigenvalues λ_i of the matrix $\boldsymbol{\Psi}_{\mathcal{L}}^{*}\boldsymbol{\Psi}_{\mathcal{L}}$, see Appendix 1 for a formal proof. If the model $\boldsymbol{\Psi}$ is sufficiently precise, the reconstruction error can be approximated as

$$\xi = \mathsf{E}\|\boldsymbol{f} - \widehat{\boldsymbol{f}}\|_2^2 \approx \sigma^2 \sum_{i=1}^{K} \frac{1}{\lambda_i}. \tag{7}$$

Then, the sensor placement problem can be stated as follows.

Problem 1 (Sensor placement problem). *Given a linear model $\boldsymbol{\Psi} \in \mathbb{R}^{N\times K}$ for a thermal distribution $\boldsymbol{f} \in \mathbb{R}^{N}$ and a number of sensors L, find the sensor placement $\mathcal{L}$ ($|\mathcal{L}| = L$) that minimizes the reconstruction error $\|\boldsymbol{f} - \widehat{\boldsymbol{f}}\|_2^2$. Namely, we aim at solving the following optimization problem,*

$$\begin{aligned} \arg\max_{\mathcal{A}} \quad & \sum_{i=1}^{K} \frac{1}{\lambda_i} \\ \text{s.t.} \quad & |\mathcal{A}| = L \\ & \mathcal{A} \subseteq \mathcal{P}\,, \end{aligned} \tag{8}$$

where λ_i are the eigenvalues of the operator $\boldsymbol{\Psi}_{\mathcal{A}}{}^{*}\boldsymbol{\Psi}_{\mathcal{A}}$.

Problem 1 can be recast as an instance of a classic combinatorial problem, the *subset selection*, that has been proven to be NP-hard [29]. Therefore, unless $\mathsf{P} = \mathsf{NP}$, there exists no algorithm that finds the optimal solution to all the instances of Problem 1 in polynomial time with respect to the size of the input. In fact, the only way to find the optimal solution is to test all the $\binom{P}{L}$ possible placements, an unfeasible approach for most of the scenarios.

Significant research efforts have been directed towards the design of an efficient algorithm with polynomial complexity

that can find a sensor placement minimizing the reconstruction error. Note that such problem is common to many disciplines, thermal management of SoCs just being one of them. Such algorithms can be usually divided in three categories: greedy algorithms, heuristics-based algorithms, and convex relaxations. The reader can find a detailed review of sensor placement algorithms for generic linear inverse problems in [16]. For the specific case of thermal management, early efforts focused on the localization of hotspots, which are localized peaks of temperature. However, such techniques are bound to fail as technology progresses, since the number and the unpredictability of hotspots are increasing. Recently, researchers refocused their efforts on studying methods to estimate the entire thermal distribution from the few collected measurements. Such methods are similar in terms of scope and approach to the ones designed for generic linear problems [16].

In this work, we propose to use FrameSense, a greedy sensor placement algorithm based on the theoretical results we described in [16]. Such an algorithm has significant advantages over the state of the art: it has theoretical guarantees and among the lowest computational complexities. The pseudo code of FrameSense is given in Algorithm 3. Note that, it is the only algorithm in the literature that has guaranteed performances in terms of MSE.

Algorithm 3. FrameSense (Sensor Placement Algorithm)

Require: Linear Model $\mathbf{\Psi}$, number of sensors L.
Ensure: Sensor locations $\mathcal{L}$.
1: Initialize the set of locations, $\mathcal{L} = \emptyset$.
2: Initialize the set of available locations, $\mathcal{N} = \{1, \ldots, N\}$.
3: Find the first two rows to eliminate, $\mathcal{S} = \arg\max_{i,j \in \mathcal{N}} |\langle \psi_i, \psi_j \rangle|^2$.
4: Update the available locations, $\mathcal{L} = \mathcal{N} \setminus \mathcal{S}$.
5: **while** $|\mathcal{S}| < N - L$ **do**
6: Find the optimal row, $i^* = \arg\min_{i \in \mathcal{L}} F(\mathcal{S} \cup i)$.
7: Update the set of removed locations, $\mathcal{S} = \mathcal{S} \cup i^*$.
8: Update the available locations, $\mathcal{L} = \mathcal{L} \setminus i^*$.
9: **end while**

4 Numerical Experiments on a 64 Cores SoC

4.1 The Experimental Setup

We test the proposed models with a real high-end many-core architecture designed for signal processing and data-intensive embedded applications that has been already taped out. This architecture hosts 64 cores designed for *multiple program multiple data* parallel computing. The cores are grouped in four clusters with independent power and clock domains and connected with a fully asynchronous network-on-chip, see Fig. 1a. The chip is implemented with STM 28 nm CMOS technology [30] and has a power density of 55 $\mathbf{W/cm^2}$.

The power traces of the SoC components are generated by running benchmarks on an instruction-level architectural simulator equipped with an accurate and detailed power model. Such power traces are generated with a time resolution of 1 ms and are successively used to generate a set of thermal distribution representing the temperature of the SoC at run-time. As a thermal simulator, we chose 3D-ICE [28], that is based on a *transient and compact thermal model*, and we tuned it for the STM technology [7], [8].

To insure that the thermal distributions match correctly the layout of the many-core architecture, the floorplan that maps the power consumption of the hardware units (cores, memories, interconnects, etc.) to the surface of the silicon die has been extracted by processing the post synthesis layout.

Moreover, to compute precise and realistic temperatures, we initially implemented a model of the chip on a commercial computational fluid dynamics program, named ANSYS CFX [31]. The purpose of this setup is to extract the correct values for the boundary conditions of the heat dissipation in a steady state worst case simulation. Once the heat transfer coefficients are obtained, the silicon die is modeled in 3D-ICE to perform the transient thermal simulations. Note that the thermal properties of materials as well as geometries of the package are taken from [32]. In this paper, we did not consider intra and inter-die variations due to process variations [33], [34]. Nonetheless, it is possible to extend our results in this direction given the existence of an accurate model [34].

While we analyzed thermal distributions originated by several workloads, in this paper we discuss the results for three fundamental ones. Such workloads are designed to represent exhaustively the thermal scenarios that can be expected by such a 64-cores SoC. We did not consider the standard SPEC benchmarks for two reasons:

- the considered architecture is not able to execute it entirely,
- it is more interesting to use workloads representing different phases, such as parallel or sequential computation, to study the thermal behavior of a many-core architecture.

The characteristics of the three datasets are summarized in Table 1 and described more in details in what follows.

The first benchmark is a parallel 64×64 matrix-matrix multiplication that distributes the load evenly among the 64 available processing units. The multiplication is repeated to generate a load of 75 ms during which a uniform and constant heat flux is produced as in the typical scenario of an extremely parallel application.

The second benchmark is a two-phases sorting algorithm run on a vector storing 16K float values. In the first phase, individual cores are activated in sequence to sort their corresponding sixteenth part of the input using the *insertion sort* algorithm. Then, in the second phase, the cores run in parallel to merge the ordered sub-vectors to get the final output, as in the *merge sort* algorithm. The number of active cores in this latter phase is halved at every iteration. The whole application is repeated on different input vectors to generate a trace of 150 ms reproducing the scenario of a parallel application with data dependencies.

The third dataset is generated by means of Poisson processes ($\lambda = 60$, $T_{\max} = 6$) bounded by the idle and maximum power consumptions of cores, memories and other hardware modules in the chip. Such a workload, while being synthetic and randomly generated, has a significant role since we show the possibility to use it to train the model and optimize the sensor placement successfully for the real data.

TABLE 1
Characteristics of the Considered Datasets for the Training of the Proposed Method and the Evaluation of Its Performance

Name of the dataset	Content	is parallel?	is realistic?	Resolution	# of thermal maps
Dataset 0	Matrix multiplication	✓	✓	20×28	3,000
Dataset 1	Merge-sort	✘	✓	20×28	3,000
Dataset 2	Random	(✓)	✘	20×28	3,000

The obtained thermal distributions are further processed for Matlab. Given the symmetry of the architecture, all the numerical simulations consider each cluster of 16 cores independently. Note that such strategy does not imply any loss of detail or precision.

4.2 Performance Comparison between the Different Approximation Models

In this section, we compare the performance of the linear model based on PCA proposed in Section 3.2 against the model based on the DCT transform on the three dataset representing the different workloads described in Section 4.1. For the DCT model, we considered an optimized version of the model proposed in [11], where the components of the model are chosen by assuming a generic low-pass profile without studying the distribution of the specific architecture. More precisely, we select the components of the DCT transform showing the average largest coefficients over the thermal distributions belonging to the chosen training set.

For each dataset, we measure the approximation error ϵ given by each model for an increasing number of parameters, $K = \{4, 6, 8, 10, 12, 14, 16\}$. The results are given in Fig. 3. We note two main facts:

- The performance achieved by the PCA is significantly better (≥ 10 dB) than the one of the DCT model. This is expected since the PCA generates an optimized model, while the DCT-based model simply selects K columns out of a given set.
- The performance gap between PCA and DCT increases with K, meaning that PCA better exploits the increase of degrees of freedom.

With this numerical experiment, we tested the capability of the proposed techniques to capture a precise low-dimensional linear model $\mathbf{\Psi}$ from a training set $\mathcal{F}$. Note that the approximation error is just one of the aspects that defines the performance of a thermal monitoring system, but it is often the critical one to have a precise thermal reconstruction.

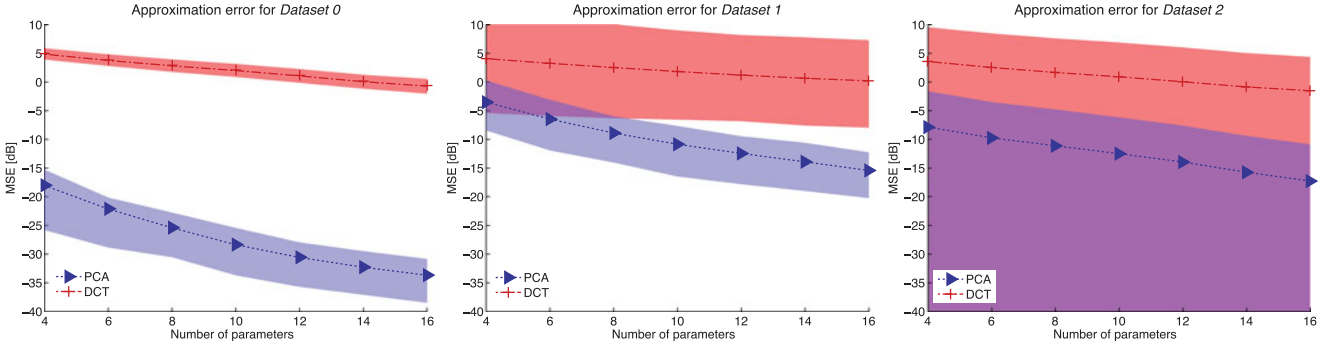
4.3 Learning Individual Models for Temperature Distributions

When learning the model $\mathbf{\Psi}$ for a certain SoC under a given workload, it is fundamental to understand how difficult it is to learn $\mathbf{\Psi}$ reliably when a part of the training set is not available. In other words, we would like to evaluate the error caused by the use of incomplete training set $\mathcal{F}$.

Consider a training set $\mathcal{F}$ of M thermal maps and define $\mathcal{A} \subseteq \mathcal{F}$ a subset of $\lfloor \delta M \rfloor$ randomly selected thermal distributions. Note that, the parameter $\delta \in [0, 1]$ represents the percentage of thermal maps that we use for training. Then, we use $\mathcal{A}$ to train the linear model and we measure the approximation error on the entire dataset $\mathcal{F}$. Note that when $\delta = 0$ it is impossible to learn the model and it is easier for increasing values of δ.

For each dataset $\mathcal{F}$, we fix K and test the performance of the PCA for a varying value of δ. We measure the performance reduction as the ratio $r(\delta)$ between the approximation error obtained while learning with the reduced training set, denoted as $\epsilon(\delta)$, and the approximation error with an entire training set, denoted as ϵ. More precisely, we define such ration as

$$r(\delta) = 10 \log_{10} \frac{\epsilon}{\epsilon(\delta)}. \tag{9}$$



(a) Matrix-vector multiplication (b) Sorting algorithm (c) Random workload

Fig. 3. Approximation errors as a function of the number of parameters K of the linear model $\mathbf{\Psi}$ for each dataset. The shaded area bounds the 5th and the 95th percentile of the approximation error of the estimated thermal distributions for each dataset. First, note that PCA is always the optimal model and the significant performance gap compared to the DCT one. This is not surprising since the DCT model has fewer degrees of freedom. Second, we underline that the difficulty of learning the model depends on the complexity of the dataset. For example, *Dataset 1* describes the thermal distributions for a sorting algorithm that cannot be implemented in a parallel fashion. Such thermal distributions are extremely heterogeneous and harder to represents with a linear model, leading to a higher approximation error compared to the other two datasets. Third, the variation for *Dataset 2* is significantly higher due to the random nature of the workload.

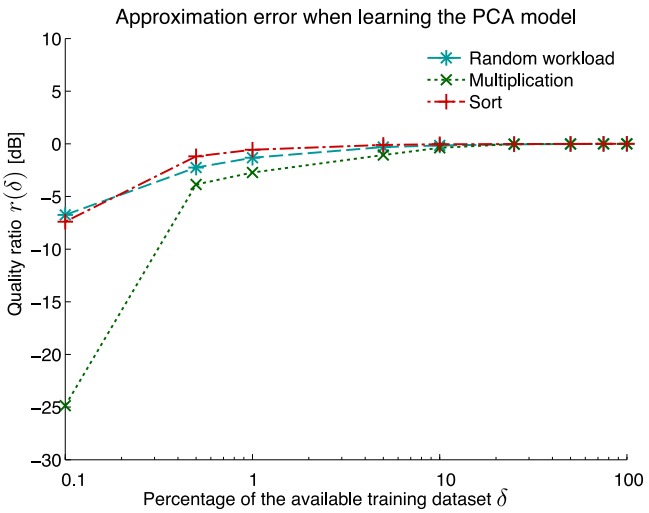


Fig. 4. Quality of the linear model learned by the PCA, when only a random part of the training set is available. We measure the quality using the cost function (9). Note that if $r(\delta) = 1$, there is no loss of precision and if $r(\delta) < 10$ the reconstruction errors have the same order of magnitude w.r.t. the model learned by an exact and complete dataset. These results indicate that the quality of the model is not severely impacted by imprecise or incomplete training sets $\mathcal{A}$.

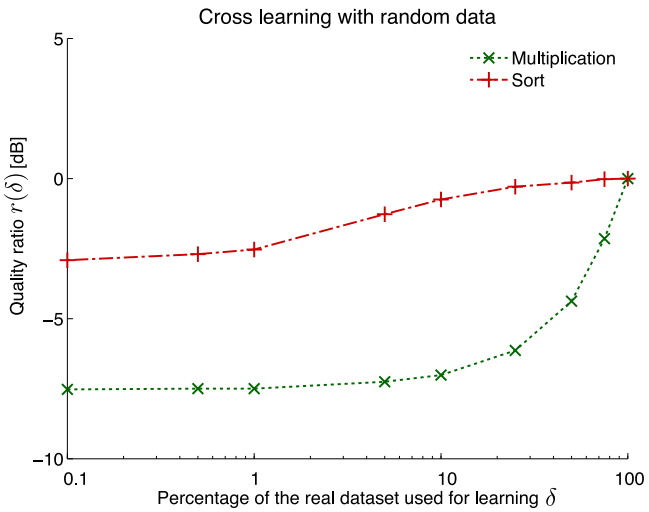


Fig. 5. Quality of the linear model learned by PCA, when only a random part of the training set is exact, and the other one is produced by random power traces. We measure the quality using the cost function (9). Note that if $r(\delta) = 1$, there is no loss of precision and if $r(\delta) < 10$ the reconstruction errors have the same order of magnitude w.r.t. the model learned by an exact and complete dataset. Even in the worst case scenario, the approximation error is of the same order of magnitude w.r.t. the optimal one.

When $r(\delta)$ is close to one, the performance of the model is not significantly impacted by the lack of data; the lower the value, the higher the sensitivity of the model to the lack of training data.

The results are given in Fig. 4, where we note that for each dataset, a subset of 1 percent of randomly selected thermal distributions is enough to have $r(\delta)$ close to one. Therefore, having a non-exhaustive dataset is in general not critical to successfully learn the PCA model. In the next section, we further strengthen our result by showing that random power traces on a realistic floorplan are sufficient to learn a reliable model Ψ.

4.4 Learning with Random Workloads

While designing an SoC and its thermal monitoring system, we may not yet know the workload. It would be then impossible to optimize the model and the sensor placement. Here, we show that actually we do not need the power-traces. In particular, we can use the randomly generated ones while maintaining reasonably good performance.

In an ideal scenario, we know exactly the thermal distributions set $\mathcal{F}$ generated by the expected workload. Assume that $\lfloor \delta M \rfloor$ thermal distributions of $\mathcal{F}$ are not known, where δ is defined as in Section 4.3. We replace the missing thermal distributions with the ones obtained using random power traces, such as the ones generated for the first dataset. We would like to measure the loss of precision of the learned model due to the increasing use of random data. Again, we use the cost function defined in (9) and the results are given in Fig. 5. While a certain loss of precision can be observed, it is extremely limited. Moreover, if we consider to train the model exclusively with the thermal distributions generated by random power-traces, that is $\delta \to 0$, then the reconstruction error is of the same order of magnitude, that is $r(\delta) < 10$. Our result indicates the possibility to learn the model and place the sensors without knowing the expected workload, it suffices to use a random one.

4.5 Reconstruction Error Comparison between Different Approximation Models

The approximation error ϵ defines the quality of the model but it is not the only merit figure. Once we place the sensors, the reconstruction error ξ may significantly vary due to the conditioning of the inverse problem (2). Therefore, we compare the different models described in Section 3.2 according to their reconstruction error for different amounts of sensors.

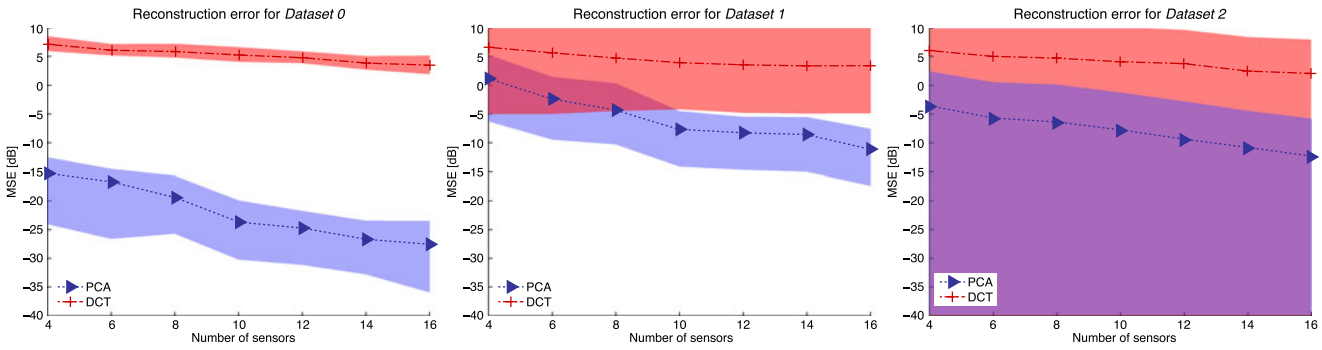
We test each model for $K = \{4, 6, 8, \ldots, 16\}$ and $L = \{4, 6, 8, \ldots, 16\}$. Note that if K decreases, it is easier to estimate the parameters but then the approximation error increases. On the other hand, if K increases it is harder to estimate the parameters $\boldsymbol{\alpha}$ but the model $\boldsymbol{\Psi}$ is more precise. The choice of the optimal K is not trivial and we perform a search over the parameter space. More precisely, for each L and each model, we measure the minimum reconstruction error obtained w.r.t. the varying model complexity K.

The results are given in Fig. 6, where we note that the PCA is the best model in terms of reconstruction error. Moreover, the advantage of the PCA w.r.t. the DCT in terms of approximation error is maintained in terms of reconstruction error. Therefore, according to our experiments, the PCA is the technique generating the model with the best approximation and reconstruction error.

4.6 Performance Comparison between Sensor Placement Algorithms

As we have already mentioned, there are many parameters impacting the performance of a thermal monitoring system, such as the number of parameters, the chosen model and the reconstruction technique. Here, we would like to compare the quality of different sensor placement algorithms while maintaining all other parameters fixed.

Therefore, we choose as a linear model the one optimized using the PCA. Then, we optimize the sensor placement



(a) Matrix-vector multiplication (b) Sorting algorithm (c) Random workload

Fig. 6. Reconstruction errors as a function of the number of sensors S of the linear model Ψ for each dataset using FrameSense as a sensor placement algorithm. The shaded area bounds the 5th and the 95th percentile of the reconstruction error for the different models. First, note that the PCA is always the optimal model and the significant performance gap compared to the DCT one. Such a gap is around or larger than 10 dB meaning that the reconstruction error of the PCA model is at least one order of magnitude smaller than the one of the DCT model. Second, we underline how homogeneous thermal distributions, such as the ones generated by the extremely parallel workload of *Dataset 0*, are easier to reconstruct.

using a few algorithms from the literature and measure the reconstruction error. For each algorithm, the reconstruction error is computed for different errors of L and K, and for each L we pick the minimum value achieved w.r.t. the number of parameters K.

We tested three algorithms:

- FrameSense, based on our theoretical results [16],
- a method based on the coherence of the measurements, that we proposed in [14] for the thermal monitoring problem,
- the information-theoretic approach described in [15], that maximizes the information collected by the sensors,
- the energy-center allocation method proposed in [11], that places the sensors where the temporal energy of the temperature distribution is higher.

Note that these algorithms have been designed to optimize the sensor placement for different scenarios and models. However, the proposed experiment is interesting to compare these algorithms when all other parameters are fixed.

The results for the three considered dataset are shown in Fig. 7, where we note that FrameSense is significantly better than the other algorithms for almost every L.

4.7 Performance Comparison between Thermal Monitoring Techniques

In the previous experiment, we compared the performance of the sensor placement algorithms when using the linear model Ψ. As we previously mentioned, the results are interesting and informative but they cannot be considered as a global measure of the different sensor placement algorithms because most of the algorithms are designed to work jointly with a specific reconstruction model, that may be significantly different from the considered linear one.

Therefore, we compare the following three thermal monitoring techniques:

- our proposed method that is based on a linear model optimized using the PCA and FrameSense as sensor placement algorithm. This is an improved version of the algorithm described in [14]; in particular, FrameSense has been proved to be theoretically near-optimal,
- the thermal monitoring approach based on spectral methods described in [11], that uses a DCT-based linear model and the energy-center algorithm previously described,
- the information-theoretic method proposed in [15], that uses the correlation matrix Σ_f and an entropy-based algorithm to recover the thermal distributions.

Each technique is composed by a sensor placement algorithm, a model for the temperature distributions and a reconstruction algorithm.

In this experiment, we vary the number of sensors $S = \{4, 5, \ldots, 23, 24\}$ and the number of parameters $K = \{4, 5, \ldots, 23, 24\}$. We added some noise to the measured values in the form of i.i.d. random Gaussian variables with variance $\sigma^2 = 4\,^{\circ}\mathrm{C}$ to simulate the presence of measurement errors. Then, for each L we picked the minimum reconstruction error achieved across all possible values of K. The results are

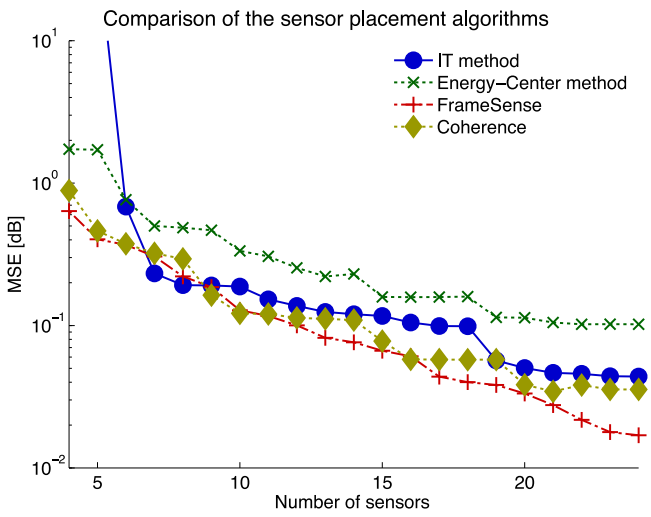


Fig. 7. Comparison of the reconstruction error between different sensor placement algorithms when Ψ is optimized using the PCA. Note that the number of parameters K is optimized by a local search independently for each value of the number of sensors L. We underline how FrameSense always generates a sensor placement that is either optimal or close to the optimal value for every L.

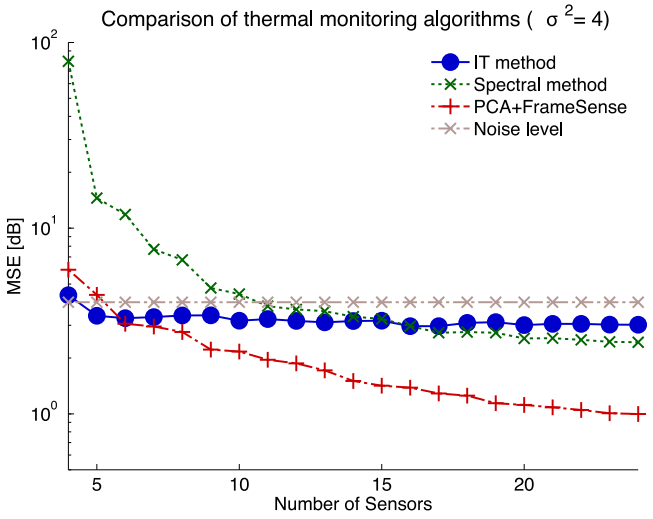


Fig. 8. Comparison of the reconstruction error between different thermal monitoring techniques when noise is perturbing the measurements. We considered an i.i.d. Gaussian noise with variance $\sigma^2 = 4^{\circ}\mathrm{C}$, a reasonable value according to the literature [23]. Note that our proposed method based on the PCA model and FrameSense as a sensor placement algorithm always achieves the lowest reconstruction error. For example, if we consider $S = 16$ sensors we reduce the reconstruction error by 42 percent when compared to the spectral method. The achieved error is even lower than the noise level, due to the regularization induced by the low-dimensional subspace $\boldsymbol{\Psi}$.

shown in Fig. 8 where we note that for a small number of sensors, i.e. $S < 10$, the performance of our proposed method and information-theoretic method are similar. However, the gap between the two methods increases with K in favor of FrameSense. Therefore, FrameSense with the PCA linear model achieves the optimal performance. Note that the spectral method is significantly worse than the other two methods, but this is not surprising given the use of a DCT basis,

TABLE 2
Computational Complexity and Memory Cost of the Thermal Reconstruction Methods

Reconstruction method	Memory cost	Comp. complexity
FrameSense	NL	$\mathcal{O}(NL)$
IT-based method [15]	NL	$\mathcal{O}(NL)$
Spectral method[1] [9]	KL or NL	$\mathcal{O}(NL)$ or $\mathcal{O}(N^3)$

[1]*The two cost refer to two opposite reconstruction strategies. The first one assumes that we store the estimation matrix given in (3); the second one assumes that we store just he coefficients and compute the matrix $\boldsymbol{\Psi}\boldsymbol{\Psi}_{\mathcal{L}}^{+}$ at run-time.*

that cannot be deeply optimized. A set of thermal maps reconstructed by the different methods is given in Fig. 9, where we considered $S = 24$ sensors, $K = 8$ parameters and a noise with variance $\sigma^2 = 4$. Note how the proposed method outperforms the others in terms of precision.

Another interesting aspect is the role of the low-dimensional model $\boldsymbol{\Psi}$ as a regularization mitigating the measurement noise. In fact, when using the PCA model and FrameSense and if the number of sensors is sufficiently high, the reconstructed thermal distributions have a lower error level w.r.t. the noise in the measurements collected by the sensors.

5 Comparison of the Computational Complexity

As a conclusive part of the numerical experiments, we would like to analyze the computational complexity and the memory cost of the different reconstruction methods. Note that we do not analyze such costs for the sensor placement algorithms because it is an off-line procedure and its costs are generally not critical.

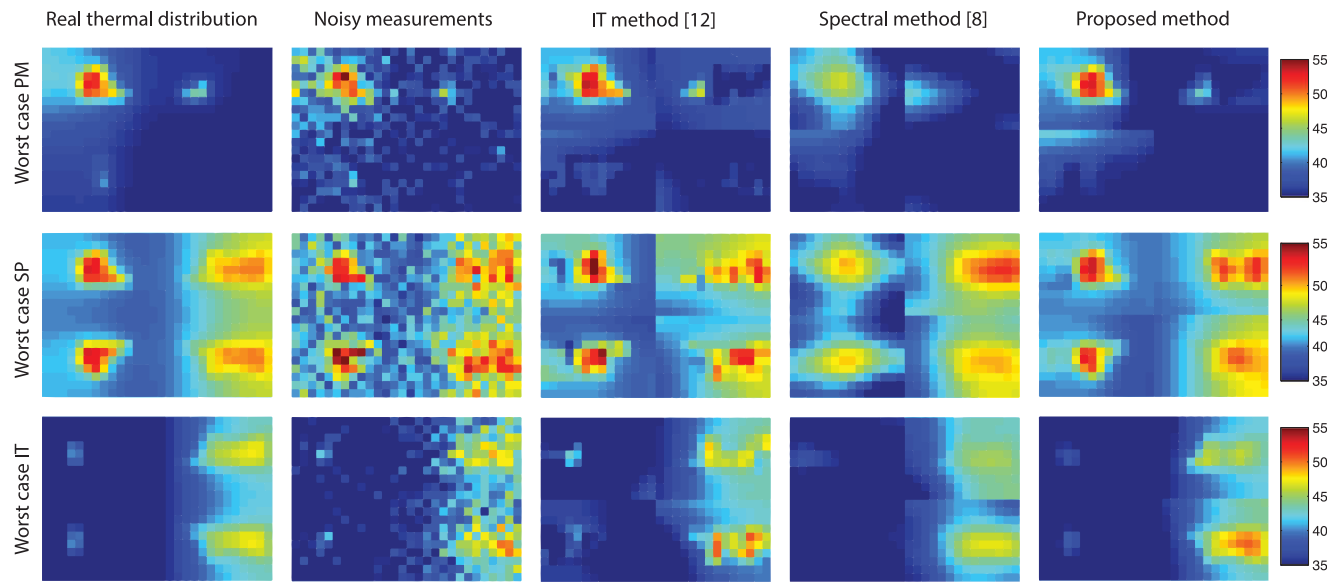


Fig. 9. Example of reconstructed thermal distributions with the different techniques. We considered $S = 24$ sensors, $K = 8$ parameters and i.i.d Gaussian noise with variance $\sigma^2 = 4$. For each row, we picked the worst-case thermal distribution for each technique. The column represent the original distribution, the noisy one and the three reconstruction with the different techniques, respectively. The colormap has been fixed and is the same for each plot; dark blue is $35\,^{\circ}C$, while red is $55\,^{\circ}C$. First, the proposed method, that is shown in the third column, reconstructs a thermal distribution that is always closer to the real one than the state of the art. Second, the spectral method tends to over-smooth the reconstructed thermal distribution due to the low-pass assumption. Third, we note how the information-theoretic method is very sensitive to noise due to the lack of regularization. As a concluding remark, the bordering effect that is often noticeable is due to the independent analysis of each cluster, as explained in Section 4.1.

The analysis is summarized in Table 2. First, we note that the only significant difference regards the memory cost of the methods. In particular, the Spectral method does not need to store an entire matrix $\boldsymbol{\Psi}_{\mathcal{L}}$ because the coefficients are usually stored in the system. However, if we choose to not to store the matrix but just the indexes of the sensor positions and of the chosen components, we need to compute the matrix multiplication (3) and the pseudo-inverse of $\boldsymbol{\Psi}_{\mathcal{L}}$ at run-time, resulting in a significantly higher computational cost.

Therefore, the benefits of the DCT models are extremely limited and we indicate the PCA model to be the optimal one for the thermal monitoring applications.

These considerations are confirmed when we look at the feasibility of a run-time implementation of the proposed algorithms in the considered architecture. Let us consider thermal maps with a resolution of 28×20 measured by 16 sensors. Then, the execution of Algorithm 1 in the P2012 many-core platform only implies less than a 5 percent overhead in the total execution time (in case of computing the thermal profile in the range of 100 ms) just using one core in P2012, thus a really negligible amount in the case of the global execution in the P2012 platform containing four 16-processor clusters [8].

As a conclusive remark, we underline that part of the computational complexity of the thermal reconstruction can be mitigated if we merge such operation within the workload optimization. We describe the details of such an idea in Appendix B.

6 Conclusions

In this work, we proposed a framework to optimally reconstruct thermal maps of many-core SoC using a small number of sensors. We defined an optimal approximation of thermal maps to reduce the number of parameters to estimate, without loosing precision. We reconstructed the thermal maps using a least square approach and we exposed the critical role of the sensor location for the quality of the thermal monitoring. We concluded proposing a greedy sensor allocation algorithm that minimizes the reconstruction error by minimizing a proxy function, namely the frame potential. The sensor placement algorithm improves the coherence-based one we previously proposed [14] and is inspired by the recent theoretical findings described in [16].

We compared the proposed method against two algorithms among the state of the art, namely the information-theoretic method [15] and the spectral method [11]. We demonstrated the higher fidelity of our reconstruction using a smaller number of sensors. We showed how the proposed reconstruction algorithm is more stable w.r.t. the noise introduced by the electronics or by sensor calibration inaccuracies, thanks to the regularization imposed by the linear model $\boldsymbol{\Psi}$.

Moreover, we investigated the challenges surrounding the learning and the optimization of the linear model $\boldsymbol{\Psi}$, one of the main critical points of the framework. We showed that a training set formed by an incomplete collection of thermal distributions is enough to learn a precise model $\boldsymbol{\Psi}$. We remarked that even a training set generated by random power traces leads to a reasonably good model. Note that such discovery has a great potential, since it allows the design of thermal monitoring systems without knowing precisely the workload of the SoC.

Appendix

A. Reconstruction Error Characterization

Proposition 1. *Consider a thermal map $\boldsymbol{f} \in \mathbb{R}^N$, a given linear model $\boldsymbol{\Psi} \in \mathbb{R}^{N \times K}$ and sensor placement $\mathcal{L}$. Then, the reconstructed thermal map is equal to*

$$\widehat{\boldsymbol{f}} = \boldsymbol{\Psi}\boldsymbol{\Psi}_{\mathcal{L}}{}^{+}(\boldsymbol{f}_{\mathcal{L}} - \boldsymbol{\epsilon}_{\mathcal{L}}) + \boldsymbol{\Psi}\boldsymbol{\Psi}_{\mathcal{L}}{}^{+}(\boldsymbol{\omega}_{\mathcal{L}} + \boldsymbol{\epsilon}_{\mathcal{L}}), \tag{10}$$

where $\boldsymbol{\omega}_{\mathcal{L}}$ is the measurement noise and $\boldsymbol{\epsilon}_{\mathcal{L}} = \boldsymbol{f}_{\mathcal{L}} - \widetilde{\boldsymbol{f}}_{\mathcal{L}}$ is the approximation error due to the linear model $\boldsymbol{\Psi}$. Assume that $\boldsymbol{\omega}_{\mathcal{L}}$ is modeled as a vector of i.i.d. Gaussian random variables with variance σ^2 and that $\boldsymbol{\Psi}_{\mathcal{L}}$ has rank K, then we can bound the reconstruction error as

$$\|\boldsymbol{f} - \widehat{\boldsymbol{f}}\|_2^2 \leq \sigma^2 \sum_{i=1}^{K} \frac{1}{\lambda_i} + \left(1 + \frac{1}{\lambda_K}\right)\|\epsilon\|^2, \tag{11}$$

where λ_i is the ith eigenvalue of the operator $\boldsymbol{\Psi}_{\mathcal{L}}{}^{*}\boldsymbol{\Psi}_{\mathcal{L}}$.

Proof. First, we note that we have two independent components in the error: the white noise generated during the measurement of the temperature and the approximation error due to the linear model $\boldsymbol{\Psi}$. Given their independence, we analyze them separately.

For the Gaussian part, we use a known result [16] to obtain the first component of (11). For the approximation error, we have to consider it twice. First, when we reconstruct the thermal map, such an approximation error is amplified by the projection onto $\mathrm{span}(\boldsymbol{\Psi})$. The worst case scenario being the approximation error aligned with the eigenvalue with λ_K, the smallest eigenvalue of $(\boldsymbol{\Psi}_{\mathcal{L}}{}^{*}\boldsymbol{\Psi}_{\mathcal{L}})$. Second, by assumption our reconstruction lies on the subspace spanned by $\boldsymbol{\Psi}$, therefore we must add $\boldsymbol{\epsilon}$, leading to the $(1 + \frac{1}{\lambda_K})$ factor. □

B. Parametric Control of the Temperature

The main possible drawback of a linear model based on the PCA is the occupation in memory and its computational cost. In fact, consider the $N \times L$ matrix $\boldsymbol{\Theta} = \boldsymbol{\Psi}(\boldsymbol{\Psi}_{\mathcal{L}})^{+}$, then we are supposed to store $\boldsymbol{\Theta}$ and to compute a matrix-vector multiplication with it to estimate the current thermal distribution. Note that N can be quite a large number, being the resolution of the estimated thermal distribution, and the cost in terms of memory and computational power is significant.

Such cost is unavoidable if our target is to actually estimate the thermal distribution. However, in most of the applications of thermal monitoring we aim at controlling of the temperature. That is, we generally aim at solving a control problem similar to the following one,

$$\begin{aligned} \max_{w} \quad & P(\boldsymbol{w}) \\ \text{subject to} \quad & \boldsymbol{f} \leq t_{\max} \\ & |\boldsymbol{D}\boldsymbol{f}| \leq d_{\max,} \end{aligned}$$

where $P(\boldsymbol{w})$ is a cost function representing the performance of the system, $\boldsymbol{w}$ is the optimized workload, $\boldsymbol{f}$ is the

temperature, $t_{\max}$ is the maximum allowed temperature and $|Df| \leq d_{\max}$ represents the maximum gradient allowed for the thermal distribution. Note that there could be more constraints and a different cost function, but such changes would not invalidate the following observations.

Note that we can generalize the temperature constraints as,

$$Bf \leq a, \tag{12}$$

where B is a linear operator that generalizes the previous constraints. Solving an optimization problem with the constraint defined as (12) clearly requires the temperature of the die, and therefore the computation and the storing of the matrix Ψ.

However, the recovery of the coefficients α is sufficient, since we can rewrite the constraint as

$$C\alpha \leq a, \tag{13}$$

where $C = B\Psi$ is a matrix containing both the low-dimensional linear model and the constraint. If the solution of the optimization problem is too expensive due to the size of the constraints, we can reduce such cost by reducing their dimensions at the cost of a reduced spatial resolution.

Acknowledgments

J. Ranieri and M. Vetterli work was supported by the ERC Advanced Grant-Support for Frontier Research-SPARSAM Nr: 247006. A. Vincenzi and D. Atienza work was supported by the Nano-Tera YINS RTD project (no. 20NA21 150939), financed by the Swiss Confederation and scientifically evaluated by the Swiss NSF, as well as by the EC FP7 STREP GreenDataNet project (no. 609000). Mr. Juri Ranieri is the corresponding author.

References

[1] S. Borkar, "Design challenges of technology scaling," *IEEE Micro*, vol. 19, no. 4, pp. 23–29, Jul./Aug. 1999.

[2] M. Pedram and S. Nazarian, "Thermal modeling, analysis, and management in VLSI circuits: Principles and methods," *Proc. IEEE*, vol. 94, no. 8, pp. 1487–1501, Aug. 2006.

[3] D. Brooks, R. Dick, R. Joseph, and L. Shang, "Power, thermal, and reliability modeling in nanometer-scale microprocessors," *IEEE Micro*, vol. 27, no. 3, pp. 49–62, May/Jun. 2007.

[4] A. K. Coskun, T. Rosing, K. Whisnant, and K. C. Gross, "Static and dynamic temperature-aware scheduling for multiprocessor SoCs," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 16, no. 9, pp. 1127–1140, Sep. 2008.

[5] W. Huang, M. R. Stan, K. Sankaranarayanan, R. J. Ribando, and K. Skadron, "Many-core design from a thermal perspective," in *Proc. Des. Autom. Conf.*, 2008, pp. 746–749.

[6] A. K. Coskun, T. S. Rosing, and K. C. Gross, "Utilizing predictors for efficient thermal management in multiprocessor SoCs," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 28, no. 10, pp. 1503–1516, Oct. 2009.

[7] L. Benini, E. Flamand, D. Fuin, and D. Melpignano, "P2012: Building an ecosystem for a scalable, modular and high-efficiency embedded computing accelerator," in *Proc. Des., Autom. Test Eur. Conf. Exhibition*, 2012, pp. 983–987.

[8] D. Melpignano, L. Benini, E. Flamand, B. Jego, T. Lepley, G. Haugou, F. Clermidy, and D. Dutoit, "Platform 2012, a many-core computing accelerator for embedded SoCs: Performance evaluation of visual analytics applications," in *Proc. Des. Autom. Conf.*, 2012, pp. 1137–1142.

[9] R. Cochran and S. Reda, "Spectral techniques for high-resolution thermal characterization with limited sensor data," in *Proc. IEEE Des. Autom. Conf.*, Jun. 2009, pp. 478–483.

[10] R. Mukherjee and S. Memik, "Systematic temperature sensor allocation and placement for microprocessors," in *Proc. 43rd ACM/IEEE Des. Autom. Conf.*, 2006, pp. 542–547.

[11] A. N. Nowroz, R. Cochran, and S. Reda, "Thermal monitoring of real processors: Techniques for sensor allocation and full characterization," in *Proc. 47th ACM/IEEE Des. Autom. Conf.*, 2010, pp. 56–61.

[12] S. Reda, R. Cochran, and A. N. Nowroz, "Improved thermal tracking for processors using hard and soft sensor allocation techniques," *IEEE Trans. Comput.*, vol. 60, no. 6, pp. 841–851, Nov. 2011.

[13] S. Sharifi and T. Š. Rosing, "Accurate direct and indirect on-chip temperature sensing for efficient dynamic thermal management," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 29, no. 10, pp. 1586–1599, Oct. 2010.

[14] J. Ranieri, A. Vincenzi, A. Chebira, D. Atienza, and M. Vetterli, "EigenMaps: Algorithms for optimal thermal maps extraction and sensor placement on multicore processors," in *Proc. Des. Autom. Conf.*, Jun. 2012, pp. 636–641.

[15] H. Zhou, X. Li, C.-Y. Cher, E. Kursun, H. Qian, and S.-C. Yao, "An information-theoretic framework for optimal temperature sensor allocation and full-chip thermal monitoring," in *Proc. Des. Autom. Conf.*, 2012, pp. 642–647.

[16] J. Ranieri, A. Chebira, and M. Vetterli, "Near-optimal sensor placement for linear inverse problems," *IEEE Trans. Signal Process.*, vol. 62, no. 5, pp. 1135–1146, Mar. 2014.

[17] Z. Chen, J. Ranieri, R. Zhang, and M. Vetterli, "DASS: Distributed adaptive sparse sensing," *submitted, IEEE Trans. Commun.*, 2013, (to appear).

[18] K.-J. Lee and K. Skadron, "Using performance counters for run-time temperature sensing in high-performance processors," in *Proc. Parallel Distrib. Process. Symp.*, 2005, p. 232.1.

[19] A. Kumar, L. Shang, L.-S. Peh, and N. K. Jha, "System-level dynamic thermal management for high-performance microprocessors," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 27, no. 1, pp. 96–108, Jan. 2008.

[20] S. J. Lu, R. Tessier, and W. Burleson, "Collaborative calibration of on-chip thermal sensors using performance counters," in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Des.*, 2012, pp. 15–22.

[21] J. Long, S. O. Memik, G. Memik, and R. Mukherjee, "Thermal monitoring mechanisms for chip multiprocessors," *ACM Trans. Archit. Code Optim.*, vol. 5, no. 2, pp. 1–33, 2008.

[22] Y. Zhang and A. Srivastava, "Adaptive and autonomous thermal tracking for high performance computing systems," in *Proc. Des. Autom. Conf.*, Nov. 2010, pp. 1–6.

[23] Y. Zhang and A. Srivastava,"Accurate temperature estimation using noisy thermal sensors," in *Proc. Des. Autom. Conf.*, 2009, pp. 472–477.

[24] E. K. Ardestani and J. Renau, "ESESC: A fast multicore simulator using time-based sampling," in *Proc. IEEE 19th Int. Symp. High Perform. Comput. Archit.*, 2013, pp. 448–459.

[25] E. K. Ardestani, F. J. Mesa-Martinez, G. Southern, E. Ebrahimi, and J. Renau, "Sampling in thermal simulation of processors: Measurement, characterization, and evaluation," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 32, no. 8, pp. 1187–1200, Aug. 2013.

[26] E. K. Ardestani, E. Ebrahimi, G. Southern, and J. Renau, "Thermal-aware sampling in architectural simulation," in *Proc. ACM/IEEE Int. Symp. Low Power Electron. Des.*, 2012, pp. 33–38.

[27] C. Lian, M. Knox, K. Sikka, X. Wei, and A. J. Weger, "Development of a flexible chip infrared (IR) thermal imaging system for product qualification," in *Proc. IEEE 28th Annu. Semiconductor Thermal Meas. Manage. Symp.*, 2012, pp. 337–343.

[28] A. Sridhar, A. Vincenzi, M. Ruggiero, T. Brunschwiler, and D. Atienza, "3D-ICE: Fast compact transient thermal modeling for 3D ICs with inter-tier liquid cooling," in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Des.*, 2010, pp. 463–470.

[29] A. Das and D. Kempe, "Algorithms for subset selection in linear regression," in *Proc. 40th ACM Symp. Theory Comput.*, Jul. 2009, pp. 45–54.

[30] ST Microelectronics, "STMicroelectronics in cooperation with Soitec makes 28nm FD-SOI CMOS process available through CMP," 2012.

[31] ANSYS, Inc., "ANSYS CFX."

[32] Z. Wan, X. Luo, and S. Liu, "Thermal stress analysis of PBGA under natural convection using a fluid-solid coupling method," in *Proc. 12th Int. Conf. Electron. Packag. Technol. High Density Packag.*, 2011, pp. 1–7.
[33] E. Kursun and C.-Y. Cher, "Variation-aware thermal characterization and management of multi-core architectures," in *Proc. IEEE Int. Conf. Comput. Des.*, 2008, pp. 280–285.
[34] E. Kursun and C.-Y. Cher, "Temperature variation characterization and thermal management of multicore architectures," *IEEE Micro*, vol. 29, no. 1, pp. 116–126, Jan./Feb. 2009.

Juri Ranieri (S'10-M'15) received both the BS and MS degrees in electronic engineering, in 2007 and 2009, respectively, from Universitá di Bologna, Italy. From September 2010 to September 2014, he was at the doctoral school, École Polytechnique Fédérale de Lausanne (EPFL), where he received the PhD degree under the supervision of Prof. Martin Vetterli and Dr. Amina Chebira. From July 2009 to December 2009, he was a visiting student at the Audiovisual Communications Laboratory (LCAV), École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland. From January 2010 to August 2010, he was with IBM Zurich to investigate the lithographic process as a signal processing problem. From April 2013 to July 2013, he was an intern at Lyric Labs of Analog Devices, Cambridge. His main research interests include inverse problems of physical fields and the spectral factorization of autocorrelation functions. He is a member of the IEEE.

Alessandro Vincenzi (S'10) received the BS and MS degrees in computer science from the University of Parma, Italy, and the University of Verona, Italy, in 2007 and 2010, respectively. In 2010, he joined the Embedded Systems Laboratory Group, École Polytechnique Fédérale de Lausanne, Switzerland, where he has been working until 2014 as a research assistant. His research interests include thermal modeling of electronic devices, as well as programming on parallel and high performances architectures. He is a member of the IEEE.

Amina Chebira (S'04-M'09) received the bachelor's degree in mathematics from University Paris 7 Denis Diderot, in 1998, and the BS and MS degrees in communication systems from the Ecole Polytechnique Fédérale de Lausanne, Lausanne, Switzerland, in 2003, and the PhD degree from the Biomedical Engineering Department, Carnegie Mellon University, Pittsburgh, PA, in 2008, for which she received the biomedical engineering research award. She is a senior research and development engineer at the Swiss Center for Electronics and Microtechnology (CSEM), Neuchâtel, Switzerland. She was a postdoctoral researcher with the Audiovisual Communications Laboratory, EPFL, from 2008 to 2012. Her research interests include frame theory and design, biomedical signal and image processing, pattern recognition, filterbanks and multiresolution theory. She is a member of the IEEE.

David Atienza (M'05-SM'13) received the MSc and PhD degrees in computer science and engineering from UCM, Spain, and IMEC, Belgium, in 2001 and 2005, respectively. He is an associate professor of electrical and computer engineering, and the director of the Embedded Systems Laboratory (ESL) at École Polytechnique Fédérale de Lausanne (EPFL), Switzerland. His research interests include system-level design methodologies for high-performance multi-processor system-on-chip (MPSoC) and low-power embedded systems, including new 2-D/3-D thermal-aware design for MPSoCs, ultra-low power system architectures for wireless body sensor nodes, HW/SW reconfigurable systems, dynamic memory optimizations, and network-on-chip design. He is a co-author of more than 200 publications in peer-reviewed international journals and conferences, several book chapters, and five U.S. patents in these fields. He has received several best paper awards and he is (or has been) an associate editor of *IEEE Transactions on Computers, IEEE Design & Test of Computers, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, and Elsevier *Integration*. He is the Technical Programme chair of IEEE/ACM DATE 2015. He received the IEEE CEDA Early Career Award in 2013, the ACM SIGDA Outstanding New Faculty Award in 2012, and a Faculty Award from Sun Labs at Oracle in 2011. He is a distinguished lecturer (2014-2015) of the IEEE Circuits and Systems Society (CASS), and a senior member of the IEEE and the ACM.

Martin Vetterli (S'86-M'86-SM'90-F'95) received the Dipl El-Ing degree from Eidgenossische Technische Hochschule (ETHZ), Zurich, in 1981, the master's of science degree from Stanford University, in 1982, and the Doctorat es Sciences degree from the École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland, in 1986. After his dissertation, he was an assistant and then an associate professor in electrical engineering at Columbia University, New York, and in 1993, he became an associate and then full professor at the Department of Electrical Engineering and Computer Sciences, University of California at Berkeley. In 1995, he joined the EPFL as a full professor. He held several positions at EPFL, including the chair of Communication Systems and the founding director of the National Competence Center in Research on Mobile Information and Communication systems (NCCR-M3S). From 2004 to 2011, he was the vice president of EPFL and from March 2011 to December 2012, he was the dean of the School of Computer and Communications Sciences. Since January 2013, he has been leading the Swiss National Science Foundation. He works in the areas of electrical engineering, computer sciences and applied mathematics. His work covers wavelet theory and applications, image and video compression, self-organized communications systems and sensor networks, as well as fast algorithms, and has led to about 150 journals papers. He is the co-author of three textbooks, with J. Kovacevic, *Wavelets and Subband Coding* (Prentice-Hall, 1995), with P. Prandoni, *Signal Processing for Communications*, (CRC Press, 2008) and with J. Kovacevic and V. Goyal, of the forthcoming book *Fourier and Wavelet Signal Processing* (2012). His research resulted also in about two dozen patents that led to technology transfers to high-tech companies and the creation of several start-ups. His work won him numerous prizes, like best paper awards from EURASIP in 1984 and of the IEEE Signal Processing Society in 1991, 1996, and 2006, the Swiss National Latsis Prize in 1996, the SPIE Presidential award in 1999, the IEEE Signal Processing Technical Achievement Award in 2001, and the IEEE Signal Processing Society Award in 2010. He is a fellow of the IEEE, the ACM and EURASIP, was a member of the Swiss Council on Science and Technology (2000-2004), and is an ISI highly cited researcher in engineering.