

# Temperature Sensor Placement Including Routing Overhead and Sampling Inaccuracies

Pablo Ituero    Fernando Garcia-Redondo    Marisa Lopez-Vallejo

**Abstract**—Dynamic thermal management techniques require a collection of on-chip thermal sensors that imply a significant area and power overhead. Finding the optimum number of temperature monitors and their location on the chip surface to optimize accuracy is an NP-hard problem. In this work we improve the modeling of the problem by including area, power and networking constraints along with the consideration of three inaccuracy terms: spatial errors, sampling rate errors and monitor-inherent errors. The problem is solved by the simulated annealing algorithm. We apply the algorithm to a test case employing three different types of monitors to highlight the importance of the different metrics. Finally we present a case study of the Alpha 21364 processor under two different constraint scenarios.

## I. INTRODUCTION

Temperature is a growing issue for electronics designers. High temperature peaks in determined places, known as hotspots, lead to unreliable circuit operation and degrade the performance of the system. Dynamic thermal management techniques dynamically adapt the circuit particular thermal conditions to increase lifetime and reliability. In this scenario, built-in temperature monitoring systems play a decisive role. However, allocating an arbitrarily large number of temperature monitors supposes a significant area and power overhead. Therefore, the number and location of thermal monitors must be optimized based on the characterization of the thermal behavior of the IC for a given application [1].

Given a set of known hotspots, finding the location of the smallest set of temperature sensors which minimizes the worst reading error is an NP-hard problem [2] and several solutions have been proposed to solve it in a computationally sensible time.

In [1], the first work of the field, a uniform and a k-means approach are presented. The former can be used if no thermal profile is available, but it generally requires a very fine-grained grill of monitors to achieve an acceptable level of accuracy. The k-means works with a fix number of k monitors it forms k clusters of hotspots and places each monitor in the thermal center of each cluster. It does not directly optimize the sensor error, but rather a combined metric of location proximity and the average temperature error. Another proposal, the RCN (Reda-Cochran-Nowroz) algorithm [2] is a heuristic algorithm composed of two phases. The first phase takes the thermal characterization data and determines the initial

location of all the sensors. The next phase iteratively improves the locations of the thermal sensors. A broader solution space is explored, achieving better results. Sensor placement has been also formulated as an integer linear program (ILP) [3]. The inputs of the program are the set of potential hotspots along with a set of the desired accuracies for these points and a set of potential sensor locations. The algorithm finds the minimum set of sensors (and their locations) that fulfills the accuracy requirements. All these proposals only target accuracy optimization.

In this paper we improve the modeling of the problem with the following elements:

- Consideration of the area and power consumption of the monitors as a design constraint.
- Consideration of the interconnection costs.
- Consideration of three inaccuracy terms: spatial errors, sampling rate errors and monitor-inherent errors.

This modeling is used by the simulated annealing algorithm to approach the problem of temperature sensor allocation. This algorithm, albeit computationally intensive, performs a very vast design space exploration finding near-optimal solutions. The algorithm yields the optimum number and location of the monitors along with the maximum allowable sampling rate.

## II. APPROACH DESCRIPTION

Figure 1 depicts an overview of the proposed temperature sensor placement and allocation environment. As shown, the algorithm receives the system layout information, the results of the thermal characterization —i.e. the hotspots information— and the area, power and accuracy constraints. Also, the algorithm retrieves the monitor and network characteristics and is

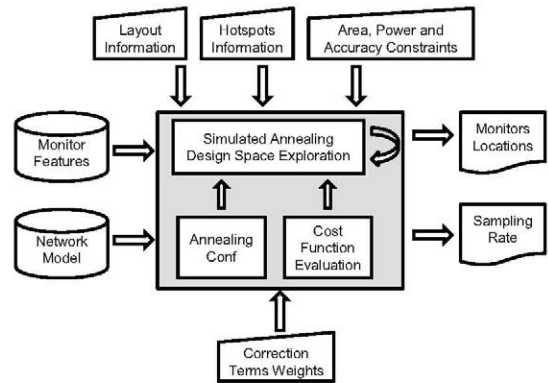


Fig. 1. Overview of the environment.

configured by means of a number of correction terms weights. The algorithm yields the location of the monitors together with the maximum allowable sampling frequency that fulfills the power budget. Next sections describe these concepts in detail.

### A. Simulated Annealing

Given a set of known hotspots, finding the location of all the elements of the smallest set of temperature sensors which minimizes the worst reading error is an NP-hard combinatorial optimization problem. This kind of problems can be solved by heuristic algorithms and by general optimization procedures.

Simulated annealing [4] is a well-known general optimization procedure that provides good solutions requiring long computation times because a vast solution space is explored. This drawback can be overtaken by making use of sophisticated cooling schedules, as is the case of the one used here [5]. Moreover, the use of incremental cost functions becomes a must to further reduce the computation time. In our case, since the allocation of sensors is a task that is carried out off-line the focus is not on the execution time of the algorithm, but on the quality of the resulting solution. Thus, a right modeling of the whole sensor allocation task becomes the definitive part of this problem. This modeling, described next, includes the definition of adequate cost metrics and the formulation of a suitable cost function.

### B. Problem Modeling

In the temperature sensor placement and allocation problem we establish the quality of a certain solution by means of four different metrics, namely the accuracy, the area, the power consumption and the interconnection costs.

The accuracy of a measurement realized in a monitor network is affected by three main error sources. First of all, monitors display a certain  $3\sigma$  error margin, referred to as  $\mathcal{E}_{sensor}$ . Secondly, the distance from the sensors to the hotspots is another source of inaccuracies. Ideally, each hotspot should have a thermal sensor placed as close to it as possible to track its temperature and deliver it to the DTM system. However, the power and area costs of temperature sensors along with their calibration overheads restrict the number of instances that can be allocated in the system. From the heat diffusion equation, we know that the difference in temperature reading of a sensor that is located at a certain distance  $r$  from a hotspot with temperature  $T_{hotspot}$  is given by:

$$\mathcal{E}_{spatial} = T_{hotspot} \left(1 - e^{-\frac{2r}{k}}\right) \quad (1)$$

where  $k$  denotes the thickness of the processor package — die, heat spreader, and thermal interface material— in terms of silicon [6]. The third source of inaccuracies is the effect of a finite sampling rate. In an ideal scenario, the thermal controller would have the readings of all the monitors constantly updated. However, due to power budget restrictions, the sampling rate of the network must be limited. This produces that the stored reading values from the monitors accumulate an error in the presence of temporary thermal gradients. Again from the heat

diffusion equation, we can establish the following expression for the error produced by this effect:

$$\mathcal{E}_{sampling} = \frac{1}{f_{sampling}} H_{max} \quad (2)$$

where  $f_{sampling}$  refers to the sampling frequency and  $H_{max}$  denotes the maximum heat flow expected on the chip. Therefore, the expression of the total deviation from the hotspot actual temperature is given by:

$$\mathcal{E}_{cost} = \mathcal{E}_{sensor} + \mathcal{E}_{spatial} + \mathcal{E}_{sampling} \quad (3)$$

The optimization of the system accuracy entails the minimization of the error committed by each monitor along with the minimization of the average error.

Concerning the area and the power consumption, the calculations are straightforward once the number of monitors,  $n$ , has been set:

$$A_{cost} = nA_{monitor} \quad (4)$$

$$P_{cost} = n f_{sampling} E_{meas} \quad (5)$$

where  $A_{monitor}$  is the area of a monitor and  $E_{meas}$  is the energy needed by a sensor to realize a measurement. These parameters are inputs of the system and depend on the characteristics of the chosen type of monitor.

For the modeling of the interconnection costs, we have considered two cases. In the former, we target a point-to-point network architecture supposing that each monitor connects to a central controller located at the weighted average of the hotspots positions and temperatures:

$$(xc, yc) = \left( \frac{\sum_i x h_i t h_i}{\sum_i t h_i}, \frac{\sum_i y h_i t h_i}{\sum_i t h_i} \right) \quad (6)$$

where  $xc$  and  $yc$  denote the coordinates of the controller and  $x h_i$ ,  $y h_i$ ,  $t h_i$  refer to the coordinates and the temperature of the  $i$ -th hotspot, respectively. The cost of the interconnection is modeled as the addition of the Manhattan distance of each monitor to the central controller:

$$I_{cost} = \sum_i (|xc - x m_i| + |yc - y m_i|) \quad (7)$$

$x m_i$  and  $y m_i$  refer to the coordinates of the  $i$ -th monitor.

The second case for the interconnection cost models a JTAG-style architecture in which the monitors are connected forming a chain of serial connections. In this case, we employ the semiperimeter method to estimate the cost:

$$I_{cost} = (\max\{x m_i\} - \min\{x m_i\}) + (\max\{y m_i\} - \min\{y m_i\}) \quad (8)$$

### C. Complete Cost Function Formulation

For our temperature monitor placement and allocation algorithm, the important information related to the process is:

- The global cost associated to the solution, provided by the area, power consumption and interconnection costs.
- The design constraints (maximum available area, power consumption and acceptable error) and goals (target average error).

This information should be considered not only in a quantitative but also in a qualitative way, taking into account its nature. In this sense, design constraints must define the search space and cost issues must be used to characterize the quality of the solution. The type of cost related to the quality attributes should also be taken into account: fixed costs must be considered in a different way than variable costs.

To build the expression of the cost function, we follow [7] where they propose the use of different correction terms (one per constraint) to guide the search in the design space.

$$\mathcal{F}(\mathcal{S}) = \sum_i k_i \times \frac{C_i(\mathcal{S})}{C_i} + \sum_i k_{c_i} \mathcal{F}_C(C_i(\mathcal{S})) \quad (9)$$

where  $\mathcal{S}$  is the solution under evaluation,  $C_i(\mathcal{S})$  is the value of a particular objective or cost,  $C_i$  is the  $i$ -th design constraint applied to the quality attribute  $C_i(\mathcal{S})$  in a given partition  $\mathcal{S}$ , the correction terms are denoted by  $\mathcal{F}_C(C_i(\mathcal{S}))$  and  $k_i$  is the weight factor for the  $i$ -th correction term.

### D. Design Space Exploration

To cover the spectrum of possible solutions as much as possible, we have employed the following set of allowed moves. During the first phase of the annealing the only change that is allowed is the addition of new sensors at random places. Once this initial solution is set up, these are the allowed movements:

- Add a new sensor at a random place.
- Remove a random sensor.
- Move a random sensor to a new random position.
- Randomly change the sampling ratio.

## III. RESULTS AND DISCUSSION

In order to emphasize the importance of the different metrics and provide a deeper understanding of the trade-offs inherent to the algorithm, in this section we first apply it to a test floorplan of  $25 \times 50 \text{ mm}^2$  containing 20 hotspots, 10 of which are spread out homogeneously in the leftmost half and reach  $85^\circ\text{C}$ , whereas the other 10 are spread likewise in the rightmost half and reach  $65^\circ\text{C}$ . For this floorplan we apply two different sets of constraints to three types of monitors. In this test case the type of monitor is fixed, therefore we can analyze the implications of having different monitor characteristics in the resulting allocation. Summarizing we apply each constraints set to the three monitors, making a total of six test scenarios.

In the first three scenarios we target minimizing (no  $k_{c_i}$  terms) the average and maximum error and set an area constraint of  $2.1 \text{ mm}^2$ , employing a penalty function. In the

TABLE I  
SELECTED MONITORS CHARACTERISTICS.

Monitor #	Source	Area( $\text{mm}^2$ )	PP Err ( $^\circ\text{C}$ )	nJ
M1	[8]	0.032	1.8	4.1E-01
M2	[9]	1.0	0.2	3.5E+03
M3	[10]	0.18	1.0	3.0E-01

TABLE II  
SUMMARY OF RESULTS.

Case	$\mathcal{E}_{cost}(^\circ\text{C})$	#Mon.	$\text{mm}^2$	mW	Samples/sec
Area-driven M1	1.90	20	0.64	8.2	1000
Area-driven M2	6.46	2	2.0	3.7E4	5345
Area-driven M3	2.47	11	1.98	3.3	10000
Power-driven M1	4.73	9	0.28	5.0	1355
Power-driven M3	3.43	10	1.8	5.0	1666
Alpha w/o net	1.69	11	1.98	3.3	10000
Alpha with net	1.80	10	1.8	3.0	10000

second three scenarios, we also target minimizing the errors but the constraint is set for a power consumption bigger than  $5 \mu\text{Watt}$ . The three monitors have been selected from [11], they all are fabricated in a  $0.18 \mu\text{m}$  technology node and feature different trade-offs between area, precision and power consumption as shown in table I.

The summary of the results of these placement/allocation is shown in table II, as shown, although we prepared six scenarios, just five feasible solutions were produced by the algorithm; the power-driven constraints were too tight for the second monitor that is very energy hungry and did not produce any outcome. Figures 2 and 3 show the resulting distributions of monitors. Figure 2 has the three first scenarios superposed whereas 3 has the remaining two. To the left of each hotspot there is a list with the maximum measurement error for that hotspot corresponding to each type of monitor distribution. Errors corresponding to type one are at the top and the ones corresponding to type three are at the bottom.

As shown, in the first three scenarios, as the power consumption was left as a secondary metric, the limit in area established the maximum number of monitors, and thus, the

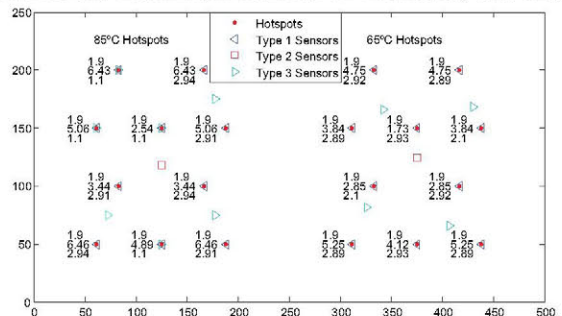


Fig. 2. Sensor distributions. Area-driven cases.

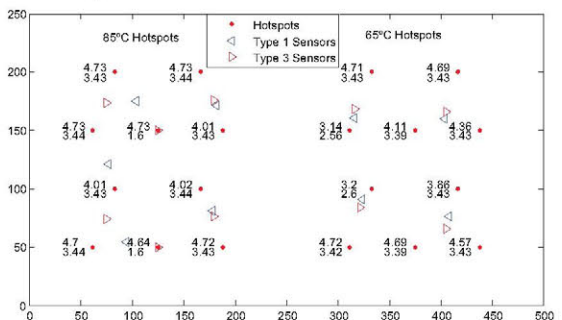


Fig. 3. Sensor distributions. Power-driven cases.



