

# Fine-Grained Clustering-Based Power Identification for Multicores

Mohamed R. Elshamy\*, Mehdi Elahi<sup>†</sup>, Ahmad Patooghy<sup>†</sup>, and Abdel-Hameed A. Badawy\*

\*Klipsch School of ECE, New Mexico State University, Las Cruces, NM 88003, United States

<sup>†</sup>Computer Systems Technology, North Carolina A&T State University, Greensboro, NC, United States

\*{elshamy, badawy}@nmsu.edu, <sup>†</sup>melahi@aggies.ncat.edu, apatooghy@ncat.edu

**Abstract**—Fine-grained power estimation in multicore Systems on Chips (SoCs) is crucial for efficient thermal management. BPI (Blind Power Identification) is a recent approach that determines the power consumption of different cores and the thermal model of the chip using only thermal sensor measurements and total power consumption. BPI relies on steady-state thermal data along with a naive initialization in its Non-negative Matrix Factorization (NMF) process, which negatively impacts the power estimation accuracy of BPI. This paper proposes a two-fold approach to reduce these impacts on BPI. First, this paper introduces an innovative approach for NMF initializing, *i.e.*, density-oriented spatial clustering to identify centroid data points of active cores as initial values. This enhances BPI accuracy by focusing on dense regions in the dataset and excluding outlier data points. Second, it proposes the utilization of steady-state temperature data points to enhance the power estimation accuracy by leveraging the physical relationship between temperature and power consumption. Our extensive simulations of real-world cases demonstrate that our approach enhances BPI accuracy in estimating the power per core with no performance cost. For instance, in a four-core processor, the proposed approach reduces the error rate by 76% compared to BPI and by 24% compared to the state of the art in the literature, namely, Blind Power Identification Steady State (BPISS). The results underline the potential of integrating advanced clustering techniques in thermal model identification, paving the way for more accurate and reliable thermal management in multicores and SoCs.

**Index Terms**—Power Estimation, Blind Power Identification, Thermal Estimation

## I. INTRODUCTION

As the pace of Moore's law decelerates, designers have sought alternative design methodologies, which led to the development of heterogeneous multicore architectures and the incorporation of specialized hardware units into a single chip, also known as Systems on Chip (SoCs) [1], [2]. However, this advancement has introduced challenges in thermal management, power consumption, and energy efficiency [1]. The increased power density coupled with constrained cooling options in SoCs has led to significant performance bottlenecks [3], [4].

Accurately estimating the power consumption of each core in a multicore processor is critical for effective thermal management and performance optimization. Precise power estimation improves the efficiency of dynamic voltage and frequency scaling (DVFS) and thermal throttling, maintaining

safe temperature limits and enhancing processor reliability and lifespan. For instance, the Running Average Power Limit (RAPL) interface allows applications to measure power consumption [5]. However, these measurements are generally coarse-grained, providing only the overall power consumption of all cores, uncore units, and total package power. Inaccurate estimations can cause thermal hotspots, suboptimal performance, reduced lifespan, and increased cooling costs due to the need for additional cooling mechanisms [6]–[8]. Therefore, it is crucial to develop techniques and tools that enable fine-grained profiling of existing SoCs and the software they operate. This step is foundational for implementing effective power and thermal management strategies and designing next generation SoCs.

Blind Power Identification (BPI) techniques are used for managing power consumption and thermal behavior in multicore SoCs in cases where detailed pre-silicon models are not available [6], [7]. These methods use statistical or machine learning approaches to infer power usage and heat generation from observable data like total power consumption and thermal sensor readings. However, BPI performance is highly dependent on the model initial conditions, leading to significant accuracy and robustness variability [6], [7], [9]. This sensitivity poses challenges in environments where precise thermal management is vital, such as SoCs used in mobile applications. The variability step from Non-negative Matrix Factorization (NMF) [10], [11] used to decompose aggregated sensor data into distinct thermal resistance and power matrices, essential for managing thermal outputs in multicore processors. However, NMF heavily relies on initial values, with traditional initialization methods often being too sensitive, resulting in inconsistent results and poor model accuracy.

This paper presents an innovative approach for initializing NMF, crucial for BPI in multicore SoCs. By employing density-based spatial clustering of applications with noise (known as the DBSCAN clustering technique) [12], [13], our approach identifies cluster centroids from dense regions in the initial thermal data to be used as starting points NMF. This approach minimizes the impact of outliers and ensures NMF is initialized with optimal points from the main dataset, thereby enhancing robustness. Furthermore, leveraging steady-state temperatures for initialization improves the accuracy of power estimation by reflecting the true thermal behavior of

The authors would like to thank the following funding agencies: NSF grants 2219679 and 2219680.

the system. Our extensive simulations demonstrate that this approach significantly improves the precision of estimated power per core while maintaining computational efficiency.

The rest of this paper is organized as follows. Sections II and III review related work and provide relevant background on thermal and power modeling. Section IV introduces our proposed approach. Section V discusses the results obtained from our simulations, comparing the accuracy and efficiency of our method against existing techniques. Finally, Section VI concludes the paper.

## II. RELATED WORK

Numerous studies have explored different techniques for modeling SoCs thermal and power aspects [6], [7], [9], [14], [15]. These approaches typically employ a standard approach, aiming to discern the state space model connecting temperature and power according to Eqn. 1.

$$T_r(k) = AT_r(k-1) + BP(k). \quad (1)$$

In this model,  $T_r(k)$  and  $P(k)$  represent vectors denoting the temperature and power levels of SoC units at time  $k$ , respectively, the matrices  $A$  and  $B$  encapsulate the physical relationship between power and temperature. While matrix  $A$  is the thermal conductance matrix that illustrates the system's natural response in the absence of power input, matrix  $B$  describes the system's forced response as a function of thermal capacitance and conductance. Both matrices  $A$  and  $B$  are square, and their dimensions correspond to the number of power sources.

The power sources align with the hardware units for which thermal measurements are available. All the previous studies [6], [7], [9], [14], [15] aim to determine the power consumption of these units. Developing state space modeling matrices allows for accurate estimation and prediction of power consumption, mirroring the precision of the available thermal measurements.

The state space model presented in Eq. 1 originates from the heat diffusion equation [16]. This equation models the interaction between power and thermal properties by incorporating factors such as thermal conductivity, material density, and specific heat capacity. The formulation of the model in Eqn. 1 involves first performing a spatial discretization of the heat diffusion equation, followed by temporal discretization [6], [7].

The difficulty in obtaining detailed power measurements has resulted in only a limited number of studies that offer valuable information about the power consumption and efficiency of various SoC hardware units under the strain of different software applications [6], [7], [9], [14], [15].

In this paper, we focus on a critical algorithm, BPI [6], [7], [9], which identifies thermal models and fine-grain power consumption of a chip using only data from thermal sensors and overall power consumption measurements. The accuracy of the algorithm estimations is aimed to be improved while maintaining the runtime. Reda *et al.* [6] proposed the first version of BPI, which utilized the fast Independent Component

Analysis (ICA) algorithm [17] to initialize NMF by providing an initial guess of the factors that enhance convergence and performance by ensuring better separation of the underlying sources. Reda *et al.* [7] proposed an enhanced BPI version with a different NMF initialization, utilizing the identity matrix for the thermal resistance matrix, as both matrices have their maximum values along the main diagonal. Said *et al.* [9] introduced BPISS, an enhancement that initializes the thermal resistance matrix with the average steady state temperatures: non-diagonal elements for stressed cores and diagonal elements for cores with similar thermal characteristics. The core steady state temperature is used if no similar cores are found. The power matrix is initialized by dividing the total power based on each core temperature ratio by the sum of all core temperatures. This method better reflects the system physical characteristics and reduces error compared to identity matrix initialization.

## III. BACKGROUND

### A. Blind Power Identification Method

The goal of the blind estimation problem is to determine the matrices  $A$  and  $B$  for a specific SoC, along with the power profiles  $P(k)$  in Eq. 1. The BPI algorithm operates in two distinct phases, the initial phase is offline learning, where the system parameters  $A$ ,  $B$ , and  $R$  are estimated using steady state measurements,  $T_s$ , and the total power at each interval,  $P_{T_s}(k)$ . This foundational phase sets the groundwork for the subsequent phase. The second phase is online learning that occurs during runtime using the runtime dataset,  $T_r(k)$ , and  $P_{T_r}(k)$ . At this stage, each core power consumption is estimated dynamically, enabling real-time adjustments and optimizations. The offline training step is required to be performed just once for each SoC to establish the modeling matrices.

To compute the  $B$  matrix in the first phase, determining the  $R$  matrix (the thermal resistance matrix) in the steady-state scenario is required [7]. In this scenario, the steady-state implies that  $T_r(k) = T_r(k-1)$ . Thus, from the equations provided:

$$T_s \approx AT_s + BP_s, \quad (2)$$

$$(I - A)T_s \approx BP_s, \quad (3)$$

$$T_s \approx (I - A)^{-1}BP_s, \quad (4)$$

$$T_s \approx RP_s. \quad (5)$$

These equations collectively demonstrate how  $T_s$  (steady-state temperature) is estimated as a function of power source inputs  $P_s$  using matrices  $A$ ,  $B$ , and  $R$ . To derive the  $R$  matrix, NMF is utilized to extract  $R$  and  $P_s$  from steady-state measurements  $T_s$  and  $P_{T_s}(k)$ . Our proposed algorithm aims to determine the optimal initialization for NMF to achieve accurate power estimation for each core,  $P(k)$ .

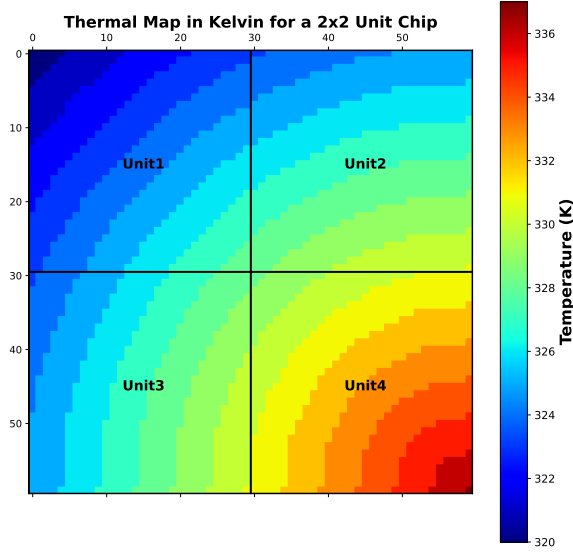


Fig. 1. A thermal map for a  $2 \times 2$  chip (in Kelvin), where the bottom right unit is activated.

#### IV. THE PROPOSED APPROACH

##### A. BPI Sensitivity to Initialization and Outliers

NMF, central to BPI, is highly sensitive to initialization and thermal outlier data. NMF's sensitivity arises from solving a non-convex optimization problem with multiple local minima, where initial values for  $R$  and  $P_s$  significantly influence the convergence point, affecting factorization quality and interpretability. Random initialization of these matrices often leads to varying results, necessitating an appropriate initialization method tailored to this specific problem [11], [18].

NMF is also sensitive to thermal outliers because outliers distort factorization, leading to poor matrix approximation and degraded feature quality. Outliers increase reconstruction error by fitting anomalous data, reducing factorization accuracy. They slow NMF convergence, requiring more iterations or causing premature convergence to sub-optimal solutions. Additionally, outliers imbalance weight distribution in factorized matrices, resulting in skewed representations of the data's underlying structure [19]. The NMF initialization is performed once during the offline phase of BPI to determine the SoC matrices  $A$  and  $B$ .

##### B. The Proposed Initialization Approach

The thermal resistance matrix,  $R$ , shows how the heat generated in one core affects the temperature of the same and other cores. As illustrated in Figure 1, a  $2 \times 2$  core chip will have a  $4 \times 4$   $R$  matrix, with the element  $r_{ij}$  showing the thermal relationship between cores  $i$  and  $j$ . When a core is activated, the highest temperature points cluster around it. Our method involves clustering thermal regions and initializing the  $R$  matrix with their centroids, which are the most influential points. From Eqn. 5,  $T_s = RP_s$ , where  $R$  directly relates to

$T_s$ , indicating that any changes in  $P$  will alter  $T_s$ . However, in a steady state, where  $T_s$  is fixed, any variation in  $R$  will correspondingly adjust the  $P_s$  matrix and vice versa.

Given  $R$ 's significant relationship with temperature, initializing the  $R$  matrix with the optimal point from the  $T$  dataset can lead to more accurate convergence of the NMF algorithm towards the true values. Additionally, every point in the original dataset impacts the elements of the  $R$  matrix. To effectively manage this, we will employ a robust clustering algorithm, DBSCAN, to categorize the original data into clusters containing the densest points and automatically eliminate outliers. This preprocessing step will aid in the initialization of the  $R$  matrix. We will also set the initial state of  $P$  to match the original steady-state temperature matrix  $T$ , allowing  $R$  to adjust  $P$  either upwards or downwards from this baseline to achieve optimal values.

DBSCAN is highly efficient in identifying clusters of various shapes and sizes within a dataset, as illustrated in Fig. 2. It typically uses the Euclidean distance to measure the distance between points:

$$d(p, q) = \sqrt{\sum_{i=1}^n (p_i - q_i)^2} \quad (6)$$

where  $p$  and  $q$  are two points in an  $n$ -dimensional space. A core point is a point  $p$  with at least  $MinPts$  points within a distance  $\epsilon$  from it (including  $p$ ):

$$|N_\epsilon(p)| \geq MinPts \quad (7)$$

where  $N_\epsilon(p)$  is the set of points within  $\epsilon$  distance from  $p$ . A point  $q$  is directly density-reachable from a point  $p$  if  $p$  is a core point and  $q$  is within the distance  $\epsilon$  from  $p$ :

$$q \in N_\epsilon(p) \quad (8)$$

A point  $q$  is density-reachable from  $p$  if there is a chain of points  $p_1, p_2, \dots, p_n$  where  $p_1 = p$  and  $p_n = q$ , and each point in the chain is directly density reachable from the previous point.

A core point has at least a minimum specified number of neighboring points ( $MinPts$ ) within a given radius ( $\epsilon$ ), central to a cluster. A border point has fewer neighbors than  $MinPts$  within  $\epsilon$  but is near a core point, typically on the cluster's periphery. Noise points do not qualify as either core or border points and are generally considered outliers. DBSCAN segments data based on point density within a specified radius, starting with a random core point and recursively exploring all density-reachable points to expand the cluster. This process continues until all potential clusters are identified. DBSCAN excels in handling noise and discovering clusters with arbitrary shapes, which many clustering algorithms struggle with. However, its effectiveness largely depends on the parameters  $\epsilon$  and  $MinPts$ , making proper selection of these parameters crucial for accurate clustering outcomes.

To estimate the optimal  $\epsilon$  for DBSCAN, the k-distance graph is used, which plots the distance to the k-th nearest neighbor for each data point in descending order. The "elbow point"

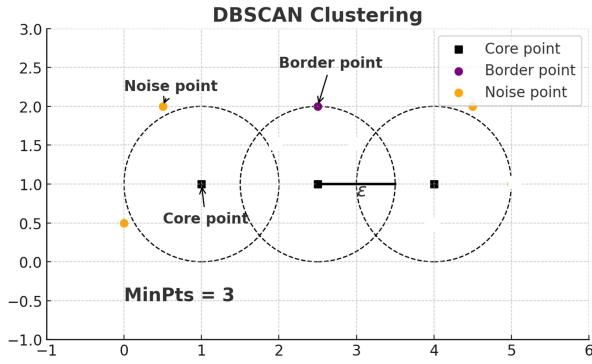


Fig. 2. DBSCAN Clustering Algorithm Illustration

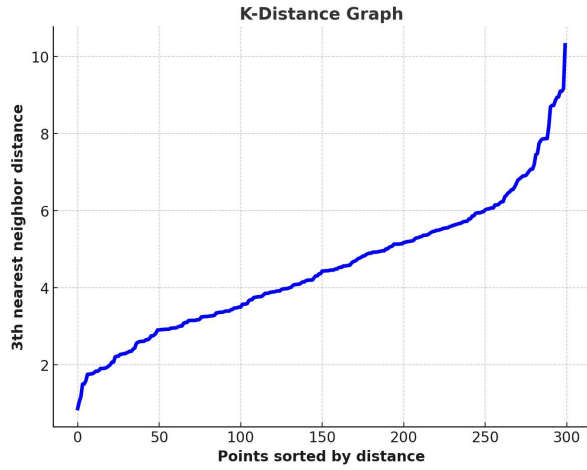


Fig. 3. K-distance graph to determine the value of  $\epsilon$

on this graph, where there is a significant change in slope, indicates the optimal  $\epsilon$ , as shown in Fig. 3. This point signifies a natural threshold between dense clusters and isolated points. The detailed structure of the proposed approach is provided in Algorithm algorithm 1.

**Algorithm 1** *Proposed Approach: Initialization with DBSCAN Clustering*

- 1: **Input:** Steady-state temperature data  $T_s$ , Number of cores  $n$ , Total power at each interval  $P_{T_s}(k)$
- 2: Determine the optimal  $\epsilon$  using the k-distance graph.
- 3: Set  $MinPts$  to at least  $D + 1$ , where  $D$  is the number of features.
- 4: Apply DBSCAN to  $T_s$  to identify clusters and remove outliers.
- 5: Initialize  $R$  with the centroids of the clusters.
- 6: Initialize  $P_s$  using  $T_s$  after outlier removal.
- 7: Perform NMF with the proposed initialization.
- 8: **Output:** System matrices  $A$ ,  $B$ , and  $R$ .

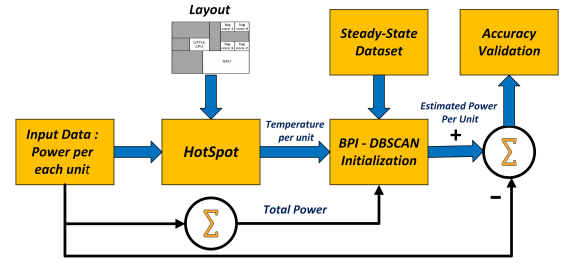


Fig. 4. The verification and testing flow of the proposed approach

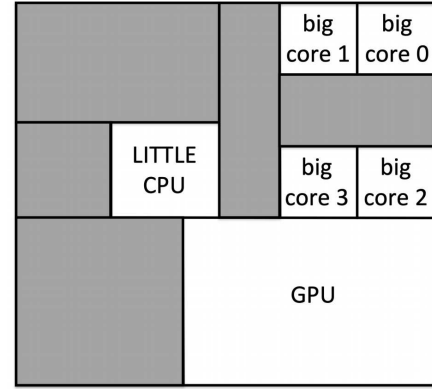


Fig. 5. The layout of the big.LITTLE+GPU SoC, utilized for testing the proposed approach [22]

## V. EXPERIMENTS AND RESULTS

### A. Experimental setup

The effectiveness of the proposed approach is validated using the HotSpot thermal simulator [20], [21] for four different floor plans described in Table I.

TABLE I  
THE FLOORPLANS WITH UNIT COUNTS, ARCHITECTURE TYPES, AND POWER BUDGETS

Floorplans	Units	Architecture	Power Budget
$2 \times 2$ mesh (FP1)	4	Homogeneous	60 W
$2 \times 4$ mesh (FP2)	8	Homogeneous	60 W
$4 \times 4$ mesh (FP3)	16	Homogeneous	60 W
Big LITTLE+GPU (FP4)	6	Heterogeneous	10 W

As depicted in Fig. 4, the HotSpot simulator receives per-unit power traces from a given design layout and generates the corresponding per-unit temperature traces. These temperature outputs are then utilized as inputs by the proposed approach alongside total power data to calculate the estimated per-unit power. Finally, the accuracy of the proposed approach is determined by comparing the initial per-unit power traces, which were inputs to the HotSpot simulator, with the power estimates generated by the proposed approach.

The proposed approach is compared with the original BPI approach, which was the first to introduce BPI for fine-grain power estimation. BPI relies on NMF to identify the  $B$  matrix. However, the accuracy of the NMF output is sensitive to the

initialization step, which in this case was performed based on the identity matrix as an initial  $R$  matrix. Therefore, it is also compared against another version of BPI [9] that improved the initialization step by utilizing steady-state temperatures, which is referred to as BPISS. Furthermore, the proposed approach is validated on a heterogeneous architecture to demonstrate that the proposed approach is robust in terms of both accuracy and computation time when transitioning from a homogeneous to a heterogeneous architecture that was depicted in Fig. 5.

### B. HotSpot Simulation Results

Figure 6 presents a comparative analysis of power estimation errors associated with two different approaches mentioned in the related work: BPI and BPISS, and the proposed approach. The analysis is carried out on four different benchmarks, detailed in Table I.

**Prediction Accuracy:** The proposed method consistently outperforms both BPI and BPISS in terms of power estimation accuracy across all tested benchmarks. In the case of FP1, the proposed approach brings the error down to around 1.2%, whereas BPI and BPISS exhibit higher errors of about 5% and 3.8%, respectively. For FP2, the error is reduced to approximately 3.5% with the proposed approach, which is notably better than the 11.5% error for BPI and 9% for BPISS. Similarly, for FP3, the error drops to around 0.9%, in contrast to the 7.5% and 6% errors seen with BPI and BPISS, respectively. Lastly, for FP4, our proposed method records an error of roughly 1%, outperforming BPI's (5.3%) and BPISS's (4.1%).

The improvement is particularly noticeable as the size of the benchmark increases, indicating the scalability of the proposed approach. The algorithm maintains a higher accuracy estimation rate when transitioning from homogeneous to heterogeneous processors, demonstrating its robustness and reliability in handling complex and larger architectures.

Figure 7 presents a comparative analysis of the runtime for the three algorithms: BPI, BPISS, and the proposed approach across the four benchmarks.

**Runtime Analysis:** The proposed approach generally shows a lower runtime compared to BPISS and is comparable to BPI in most benchmarks and helps reduce the overall computational complexity.

**Improvement over Traditional Methods:** Traditional BPI methods exhibit slightly lower run times but suffer from higher estimation errors, indicating a trade-off between speed and accuracy. BPISS shows the highest runtime, increasing computational overhead.

**Scalability and Performance:** Our proposed approach scales well with increasing benchmark sizes, maintaining efficient runtimes while handling larger and more complex architectures. This scalability ensures that the algorithm can be applied to more complex systems without a significant increase in computational resources or time.

### C. Comparison of Actual Power with Estimated Power

As depicted in Figure 8(a), illustrates the thermal measurements for each of the cores over time for heterogeneous

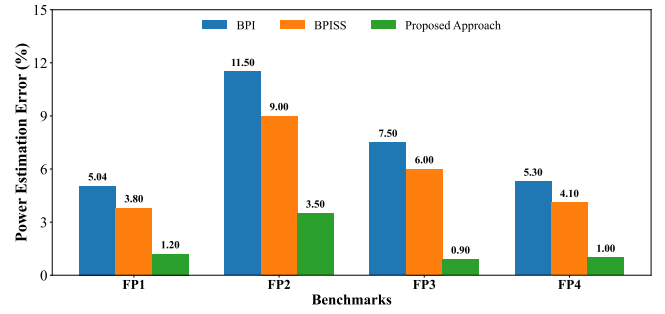


Fig. 6. Comparison of Power Estimation Errors Across Different Algorithms and Benchmarks

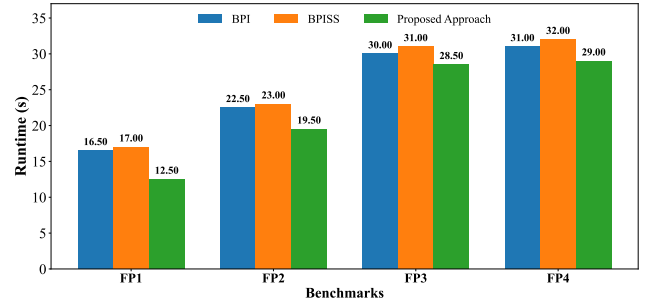


Fig. 7. Runtime Comparison of Different Algorithms Across Various Benchmarks

processors (FP4). The temperatures fluctuate as the cores experience varying states of stress and periods of idleness. These temperature variations are crucial for understanding the thermal dynamics and behavior of the processors under different operational conditions.

Figures 8 provide a detailed comparison between the power estimates derived from our proposed approach and the actual power input to each core as simulated by HotSpot. This comparison is essential to validate the accuracy and reliability of the proposed approach in estimating power consumption. The results demonstrate that the proposed approach closely matches the estimated and actual power values. This close correlation underscores the effectiveness of our approach in providing accurate power estimates, which is vital for optimizing power management and improving the overall efficiency of the SoCs.

## VI. CONCLUSIONS

Our approach markedly improves the accuracy and robustness of the Blind Power Identification (BPI) technique by employing DBSCAN for the optimal initialization of Non-negative Matrix Factorization (NMF), thereby mitigating its sensitivity to variations in datasets and outliers. Our comprehensive simulations have shown that the proposed approach provides superior accuracy in fine-grain power estimation across different multicore processor floorplans, including both homogeneous and heterogeneous architectures. Specifically, error rates in a four core processor were reduced by 67%



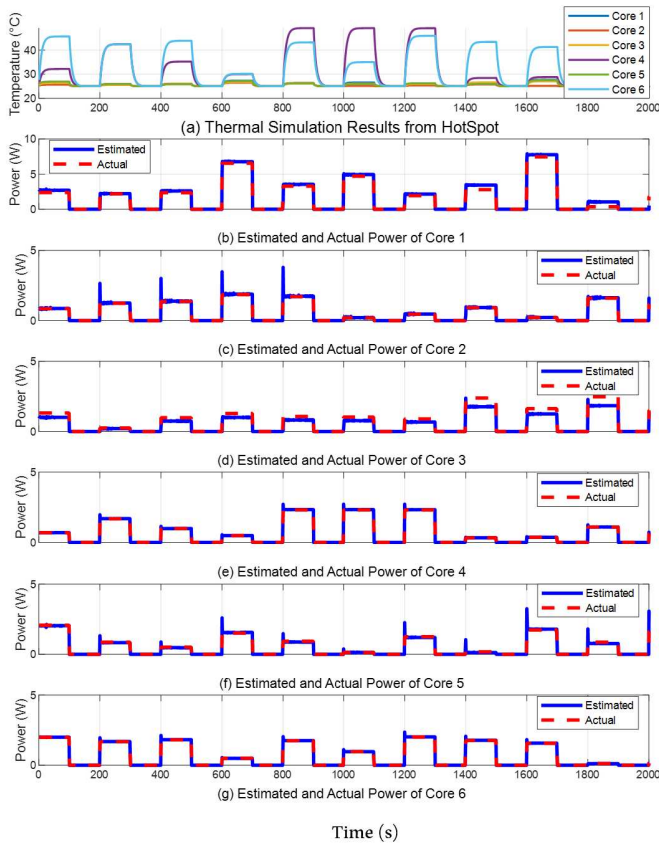


Fig. 8. The proposed BPI algorithm was validated using the HotSpot simulator based on the heterogeneous floorplan

compared to conventional BPI techniques and by 24% compared to the BPISS method. Future work could focus on integrating the proposed approach with thermal management and power control techniques, such as dynamic voltage and frequency scaling (DVFS) and thermal throttling, to create a comprehensive solution for managing power and thermal behavior in SoCs. Additionally, our approach can be combined with security techniques to identify malicious attacks based on thermal behavior.

#### ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers. This work has been partially funded by NSF grants 2219679 and 2219680.

#### REFERENCES

- [1] Jinchao Chen, Pengcheng Han, Ying Zhang, Tao You, and Pengyi Zheng. Scheduling energy consumption-constrained workflows in heterogeneous multi-processor embedded systems. *Journal of Systems Architecture*, 142:102938, 2023.
- [2] Qiangqiang Jiang, Xu Xin, Libo Yao, and Bo Chen. Metism: Multiobjective energy-efficient task scheduling model for an edge heterogeneous multiprocessor system. *Future Generation Computer Systems*, 152:207–223, 2024.
- [3] Kaige Yan, Xingyao Zhang, and Xin Fu. Characterizing, modeling, and improving the qoe of mobile devices with low battery level. In *2015 48th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO)*, pages 713–724, 2015.

- [4] Srikanth Rangarajan, Scott N. Schiffres, and Bahgat Sammakia. A review of recent developments in “on-chip” embedded cooling technologies for heterogeneous integrated applications. *Engineering*, 26:185–197, 2023.
- [5] Howard David, Eugene Gorbato, Ulf R. Hanebutte, Rahul Khanna, and Christian Le. Rapl: Memory power estimation and capping. In *2010 ACM/IEEE International Symposium on Low-Power Electronics and Design (ISLPED)*, pages 189–194, 2010.
- [6] Sherief Reda and Adel Belouchrani. Blind identification of power sources in processors. In *Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 2017, pages 1739–1744, 2017.
- [7] Sherief Reda, Kapil Dev, and Adel Belouchrani. Blind identification of thermal models and power sources from thermal measurements. *IEEE Sensors Journal*, 18(2):680–691, 2018.
- [8] Ping Fu, Lan Zhao, Xuguang Wang, Jian Sun, and Zhicheng Xin. A review of cooling technologies in lithium-ion power battery thermal management systems for new energy vehicles. *Processes*, 11(12), 2023.
- [9] Mostafa Said, Sofiane Chetoui, Adel Belouchrani, and Sherief Reda. Understanding the sources of power consumption in mobile socs. In *2018 Ninth International Green and Sustainable Computing Conference (IGSC)*, pages 1–7, 2018.
- [10] Yu-Xiong Wang and Yu-Jin Zhang. Nonnegative matrix factorization: A comprehensive review. *IEEE Transactions on Knowledge and Data Engineering*, 25(6):1336–1353, 2013.
- [11] Jiangzhang Gan, Tong Liu, Li Li, and Jilian Zhang. Non-negative Matrix Factorization: A Survey. *The Computer Journal*, 64(7):1080–1092, 07 2021.
- [12] Sebastian Raschka, Yuxi Liu, and Vahid Mirjalili. *Machine Learning with PyTorch and scikit-learn*. Packt Publishing Ltd., 3rd edition edition, 2022.
- [13] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, KDD’96*, page 226–231. AAAI Press, 1996.
- [14] Andrea Bartolini, Matteo Cacciari, Andrea Tilli, and Luca Benini. A distributed and self-calibrating model-predictive controller for energy and thermal management of high-performance multicores. In *2011 Design, Automation & Test in Europe*, pages 1–6, 2011.
- [15] Yefu Wang, Kai Ma, and Xiaorui Wang. Temperature-constrained power control for chip multiprocessors with online model estimation. *SIGARCH Comput. Archit. News*, 37(3):314–324, jun 2009.
- [16] A. Munier, J. R. Burgan, J. Gutierrez, E. Fijalkow, and M. R. Feix. Group transformations and the nonlinear heat diffusion equation. *SIAM Journal on Applied Mathematics*, 40(2):191–207, 1981.
- [17] Seyed Mohammad Reza Shahshahani and Hamid Reza Mahdiani. Fica: A fixed-point custom architecture fastica for real-time and latency-sensitive applications. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 30:2896–2905, 2022.
- [18] Flavia Esposito. A review on initialization methods for nonnegative matrix factorization: Towards omics data experiments. *Mathematics*, 9(9), 2021.
- [19] S. Fathi Hafshejani and Z. Moabferdard. Initialization for non-negative matrix factorization: a comprehensive review. *International Journal of Data Science and Analytics*, 16(2):119–134, 2023.
- [20] Wei Huang, S. Ghosh, S. Velusamy, K. Sankaranarayanan, K. Skadron, and M.R. Stan. Hotspot: a compact thermal modeling methodology for early-stage vlsi design. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 14(5):501–513, 2006.
- [21] Kevin Skadron, Mircea R. Stan, Wei Hu, Srinivasan Velusamy, Kailash Sankaranarayanan, and David Tarjan. Temperature-aware microarchitecture: Modeling and implementation. In *Proceedings. 30th Annual International Symposium on Computer Architecture*, 2003., pages 2–13. IEEE, 2004.
- [22] Young Ho Gong, Jae Jeong Yoo, and Sung Woo Chung. Thermal modeling and validation of a real-world mobile ap. *IEEE Design and Test*, 35(1):55–62, February 2018.