IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 8 November 2023, accepted 20 November 2023, date of publication 23 November 2023, date of current version 29 November 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3336280

RESEARCH ARTICLE

3D-DNaPE: Dynamic Neighbor-Aware Performance Enhancement for Thermally Constrained 3D Many-Core Systems

MOHAMMED SULTAN MOHAMMED [1,2], (Member, IEEE), AHLAM AL-DHAMARI [2], MOSAB HAMDAN [3], (Senior Member, IEEE), ABDUL-MALIK H. Y. SAAD [4], (Senior Member, IEEE), ANTAR S. H. ABDUL-QAWY [5], AND M. N. MARSONO [1]

[1]Department of Electronic and Computer Engineering, Faculty of Electrical Engineering, Universiti Teknologi Malaysia, Johor Bahru 81310, Malaysia
[2]Department of Computer Engineering, Faculty of Computer Science and Engineering, Hodeidah University, Al Hudaydah, Yemen
[3]Interdisciplinary Research Center for Intelligent Secure Systems, King Fahd University of Petroleum and Minerals, Dhahran 31261, Saudi Arabia
[4]College of Engineering, University of Buraimi, Buraimi 890, Oman
[5]Department of Mathematics and Computer Science, Faculty of Science, Abdulrahman Al-Sumait University, Zanzibar 1933, Tanzania

Corresponding authors: Mohammed Sultan Mohammed (samohammed@utm.my), Ahlam Al-Dhamari (ahlam.aldhamari@hoduniv.net.ye), and M. N. Marsono (mnadzir@utm.my)

This work was supported by the Research Management Centre (RMC), Universiti Teknologi Malaysia (UTM), under the Professional Development Research University grant (R.J130000.7113.06E45).

ABSTRACT The continuous scaling of silicon technology has enabled many-core systems to become ubiquitous, offering enormous computational power for various applications spanning from high-performance computing to mobile devices. However, this advancement resulted in increased power density that exacerbated the thermal challenges of dark silicon, where certain cores are turned off or become *dark* due to thermal constraints. While various methods have been put forward to enhance the performance of thermally constrained 2D many-core systems, 3D designs introduce more serious thermal issues due to heightened power density and challenges with heat dissipation in vertically stacked configurations. This paper introduces a dynamic neighbor-aware performance enhancement for thermally constrained 3D many-core systems (3D-DNaPE). 3D-DNaPE is a technique that improves the performance of a thermally constrained 3D many-core system where only a limited number of cores can be activated. Initially, it uses the proposed neighbor-aware pattern (NaP) algorithm to select the coldest core among the four adjacent dark cores suitable for task migration. Subsequently, it uses the proposed 3D dynamic thermal management (3D-DTM) algorithm to optimize system performance by considering the core and memory bank temperatures. A static non-uniform cache access (S-NUCA) configuration mitigates cache misses resulting from task migration. Comprehensive evaluations indicate that 3D-DNaPE performs better than its contemporaries, showing improvements reaching up to 43% in execution time, a 34% decrease in performance slowdown, and an up to 51% enhancement in energy efficiency. This research not only underscores the challenges faced by 3D many-core systems but also provides a robust solution with promising implications for future 3D many-core designs.

INDEX TERMS 3D-stacked, dark silicon, many-core system, neighbor-aware, performance enhancement, thermally constrained.

I. INTRODUCTION

The continual scaling of silicon technology in recent years has given rise to the emergence of many-core systems, which integrate many processor cores onto a single chip. These systems offer great computational power and have become the driving force behind various applications, encompassing a broad spectrum from high-performance computing to mobile devices [1]. However, this increase in computational power has also led to an increase in power density, which

The associate editor coordinating the review of this manuscript and approving it for publication was Mario Donato Marino.

© 2023 The Authors. This work is licensed under a Creative Commons Attribution 4.0 License.
For more information, see https://creativecommons.org/licenses/by/4.0/

has caused significant thermal challenges. One of these challenges is the *dark silicon* issue, which refers to the portion of cores that cannot be fully utilized due to thermal constraints [2]. In this paper, the terms *dark silicon* and *thermally constrained* are used interchangeably. The dark silicon issue is further exacerbated by the transition from 2D to 3D many-core architectures, a move aimed at overcoming off-chip memory bandwidth limitations. In these 3D architectures, cores and main memory/cache layers are vertically stacked [3], leading to even higher power density and decreased heat dissipation capabilities when layers are active [4], [5], [6], [7]. Consequently, 3D many-core systems face more significant thermal challenges compared to their 2D counterparts. Moreover, external environmental conditions, such as ambient temperature, play a pivotal role in influencing the system's thermal behavior. Additionally, within the many-core system itself, distinct zones or sections can exhibit varied thermal behaviors, necessitating targeted thermal management strategies for each zone.

Several techniques have been suggested to cope with the challenge of dark silicon or thermally constrained many-core systems, with a predominant focus on the performance improvement of 2D thermally constrained many-core systems. Some of these 2D optimization techniques concentrate on mapping and pattern techniques to enhance the performance of 2D thermally constrained many-core systems [8], [9]. Another set of techniques involves the use of the computation sprinting mechanism, which temporarily boosts the frequencies of cores utilizing dynamic voltage and frequency scaling (DVFS) [10], [11], [12], [13], [14], [15], [16]. Other techniques [17], [18], [19] leverage the dark cores to aggressively lower the system's temperature by migrating the tasks from the active cores to these dark cores and turning off the active cores. In addition, these techniques use DVFS to progressively decrease the system temperature. However, all the 2D optimization techniques only focus on the temperature of cores and ignore the temperatures of memory. On the other hand, only a few methods have been suggested to improve the performance of 3D many-core system [20], [21], [22]. However, some use a fixed power budget, ignoring transient temperature fluctuations and heat transfer across cores. Others depend on the applications' performance models being available at design time. Therefore, it cannot be used for unknown applications. Moreover, to the best of our knowledge, no one has proposed task migration in a 3D dark silicon many-core system.

This study presents a dynamic neighbor-aware performance enhancement for thermal-constrained 3D many-core systems (3D-DNaPE). The proposed technique comprises two stages. The first stage utilizes the proposed neighbor-aware pattern (NaP) algorithm to select one coldest core of the four adjacent dark cores suitable for task migration. This allows 3D-DNaPE to selectively perform task migration only for the hot cores rather than migrating all cores as was in our previously proposed DTaPO [17] for 2D dark silicon many-core. In the second stage, a 3D dynamic thermal management (3D-DTM) algorithm is used. This algorithm utilizes task migration to enhance the performance of many-core system while ensuring that the operating temperature remains within safe thermal limits. However, unlike DTaPO, which focuses solely on core temperatures, 3D-DNaPE considers both core and memory bank temperatures in 3D many-core architectures. In case there is no surrounding cold core, DVFS is used to progressively reduce the system temperature.

It is known that using task migration leads to cache misses. To address this, a shared last-level cache (LLC) can be used to mitigate the cache misses resulting from task migration [17], [23]. In DTaPO, the tasks were only migrated horizontally among the two adjacent cores that shared the same L3 cache. In contrast, the 3D-DNaPE technique allows tasks to be moved in all directions among the four adjacent cores, based on the coldest neighboring core. To enable this, a static non-uniform cache access (S-NUCA) [24] configuration is utilized as the LLC. In the S-NUCA architecture, the LLC banks are physically distributed across all cores within the many-core system. However, they still logically form a singular and vast cache shared by all cores. Such architecture can be found in commercially many-core processors [25]. During task migration in an S-NUCA many-core system setup, only the cache lines from the source core's private caches, from which the task is migrating, must be flushed to the LLC. Subsequently, the core to which the migrated task relocates can access these cache lines via the shared LLC. Thus, this strategy effectively reduces the overhead associated with task migration stemming from cache misses. In summary, the key contributions of this paper can be outlined as follows:

- We introduce a performance enhancement technique specifically designed for thermally constrained 3D many-core systems.
- We propose a neighbor-aware pattern (NaP) algorithm that selects the coldest core among the four adjacent dark cores suitable for task migration.
- We develop a 3D dynamic thermal management (3D-DTM) algorithm that leverages both task migration and DVFS to dynamically manage thermal conditions, taking into consideration the temperature profiles of cores and memory banks.
- We validate the efficacy of our proposed 3D-DNaPE technique through an exhaustive evaluation utilizing both compute- and memory-intensive multi-threaded applications.

The subsequent sections of this paper are organized as follows: Section II discusses related work. Section III presents the system model and problem definition. Section IV describes the proposed work's methodology, while Section V evaluates its performance. Lastly, Section VI summarizes

the final remarks and provides insights into future research directions.

II. RELATED WORK

The dark silicon problem raises a vital question: how can we use available computational resources effectively in the face of power and thermal limitations? This issue has gained significant attention in computer architecture and design. Researchers and engineers are searching for innovative methods to enhance the performance of many-core systems within these power and thermal limitations.

Recent years have seen several studies improving the performance of thermally constrained many-core systems. Some have employed mapping and pattern techniques [8], [9], [26], [27], while others have harnessed the computation sprinting mechanism, briefly increasing core frequencies using DVFS [10], [11], [12], [13], [14], [15], [16]. Kanduri et al. [28] introduced adBoost, a thermal-aware performance-boosting technique that patterns dark cores among active ones to create thermal headroom. Raghunathan and Garg [29] developed a scheduler using queuing theory and job arrival rates to make run-time decisions for task and cluster optimization. Mohammed et al. [17], [18] introduced a dynamic thermal-aware performance optimization technique that uses task migration and DVFS to enhance the performance of thermally constrained many-core systems. Moreover, in [30], the researchers proposed a prediction-based early wake-up of dark cores to reduce the dark cores' wake-up latency and improve the overall performance of thermally constrained many-core systems. Several other researchers have also proposed techniques emphasizing dynamic power budgeting [31], [32], [33].

All aforementioned techniques target thermally constrained 2D many-core systems. However, 3D many-core systems face more significant thermal challenges compared to their 2D counterparts. This is primarily due to higher power density and reduced heat dissipation when active layers are stacked vertically. Moreover, in addition to the cores, non-core components, such as memories and caches, play a significant role in generating heat [20], [22]. Several task scheduling-based techniques for dynamic temperature management in 3D many-core systems have been proposed [34], [35], [36], [37], [38]. The authors of [39] and [40] proposed performance optimization techniques under power and thermal constraints. Thermal management and performance optimization techniques for 3D many-core systems with hybrid SRAM/MRAM L2 caches were proposed by Lee et al. [41], [42]. By considering thermal-induced stress, Zou et al. [43] introduced a thermal managing approach for 3D systems. Wang et al. [44] proposed an artificial neural network-based run-time stress estimator. Also, STREAM was presented to optimize the 3D many-core performance considering the thermal-induced reliability issues [45]. However, the techniques above were not aimed at thermally constrained 3D many-core systems, where only a part of the system's units can be active simultaneously.

There are currently limited techniques to improve the performance of thermally constrained 3D many-core systems [20], [21], [22]. Asad et al. [20] consider the power consumption of cores and non-core components concurrently to enhance the performance of thermally constrained 3D many-core systems. However, they use a fixed power budget, which over-constrains the system's performance at run-time. Wan et al. [21] proposed a greedy-based core-cache co-optimization algorithm to optimize the performance of thermally constrained 3D many-core systems at run-time. However, it depends on the applications' performance models being available at design time. Therefore, it cannot be used for unknown applications.

Siddhu et al. [22] proposed a dynamic thermal management approach called CoreMemDTM. CoreMemDTM is a joint approach to managing the thermal levels of a computing system's processor cores and memory. Based on the idea that the core and memory are interdependent, a dynamic thermal management (DTM) decision made for one can reduce the temperature of the other, thereby lowering overheads. CoreMemDTM does this by utilizing a multi-level slack-balanced DVFS technique to control the cores (CoreDTM) and low-power states to manage the memory (MemDTM). CoreMemDTM activates an appropriate DTM policy if the temperature of the core or memory rises. When both the core and memory components overheat, CoreMemDTM executes DTM for the component with the lowest thermal slack. However, CoreMemDTM uses only DVFS and power gating and does not use task migration. Our previous work [17] has shown that task migration can substantially lower chip temperature without compromising overall system performance. This approach balances thermal loads across the cores, allowing for efficient utilization of system resources while adhering to thermal constraints.

In summary, most previous performance optimization techniques for thermally constrained many-core systems targeting 2D are not suitable for 3D stacked layers, where they only concentrate on the temperatures of cores and ignore the temperatures of memory. On the other hand, 3D system performance optimization techniques do not target dark silicon problems, where only part of a many-core system can be activated at the same time. Only a few works are aimed at thermally constrained 3D many-core systems. However, some use a fixed power budget, ignoring transient temperature fluctuations and heat transfer across the cores. Others depend on the applications' performance models being available at design time. Therefore, they fail to accommodate unknown applications that are not characterized at design time. Moreover, to the best of our knowledge, no one has used task migration in a 3D dark silicon many-core system. Our previous work [17] proves that using task migration can aggressively decrease a chip's temperature while getting good overall performance from a thermally constrained many-core system.

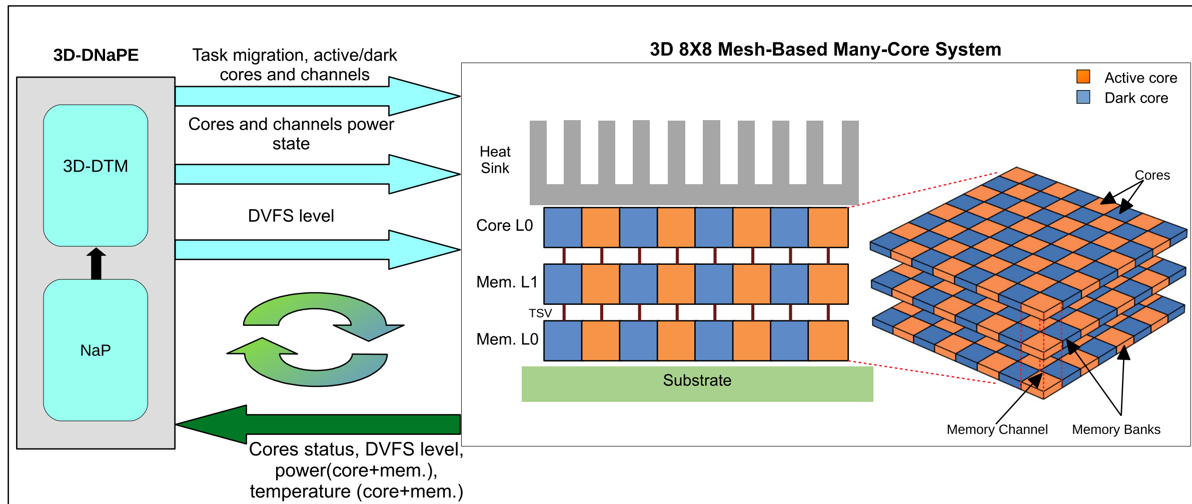

FIGURE 1. An illustration of the proposed system model.

III. SYSTEM AND APPLICATIONS OVERVIEW

This section describes the proposed system model for a 3D dark silicon many-core system, the applications under consideration, and the problem definition and formulation.

A. PROPOSED SYSTEM MODEL

This work focuses on addressing the thermal challenges and performance enhancement in 3D many-core systems. The proposed system model shown in Fig. 1 is used for evaluating the effectiveness of the proposed techniques.

While the specific configuration may not directly reflect the exact architectures available in the market, the research provides insights into the challenges and potential solutions for thermally constrained 3D many-core systems. The 3D-stacked many-core system consists of three layers: one core layer and two memory layers. The core layer comprises 64 homogeneous cores, while the memory layers consist of 128 memory banks. An 8×8 mesh-based network-on-chip (NoC) is utilized as a communication medium. The memory layers are vertically stacked and interconnected through vertical channels. These channels consist of pathways utilizing through-silicon vias (TSVs) [46], facilitating data transfer between the memory layers and the cores. As this study targets to improve the performance of the thermally constrained many-core system, we assume that only half of the cores can be activated simultaneously. Previous studies [17], [18] show that the use of half of the cores in a thermally constrained environment can give better results than using all of the cores. Initially, the active and dark cores and memory channels are organized in a checkerboard pattern, where dark cores surround each active core to enhance heat dissipation by providing thermal headroom [47]. However, this pattern keeps changing during the execution time according to the temperature of the cores using task migration. More details on how task migration changes the active and dark core patterns are discussed in Section IV.

The proposed 3D-DNaPE comprises two stages. The first stage aims to identify the coldest neighboring core to which tasks can be migrated using the neighbor-aware pattern (NaP) algorithm. The second stage involves performing DTM on the thermally constrained 3D many-core system, taking into account the individual temperatures of cores and memory banks, facilitated by the 3D-DTM algorithm. It is necessary to monitor the temperatures of cores and memory banks separately due to their varying heat dissipation characteristics, which are influenced by the specific applications being run. More details about the proposed algorithms are presented in Section IV. 3D-DNaPE continuously monitors the status of the many-core system at predefined control intervals while multi-threaded applications are running. More details about these multi-threaded applications are provided in the following subsection. Specifically, 3D-DNaPE tracks the locations of active and dark cores, the DVFS level, the power consumption of both cores and memory banks, as well as the transient temperatures of these components. Assuming that the many-core system supports preemptable tasks, 3D-DNaPE intervenes when potential thermal violations are detected. It halts the tasks and relocates them to another core selected by the first stage for continued execution. It modifies the voltage/frequency level by utilizing DVFS if no thermal headroom is available.

B. MULTI-THREADED APPLICATIONS

We focus on multi-threaded applications, drawing from a range of scientific computing and engineering domains. These applications are not inherently designed with strict real-time constraints. They are represented in the SPLASH-2 [48] and PARSEC [49] benchmark suites. The SPLASH-2 suite encompasses multi-threaded applications

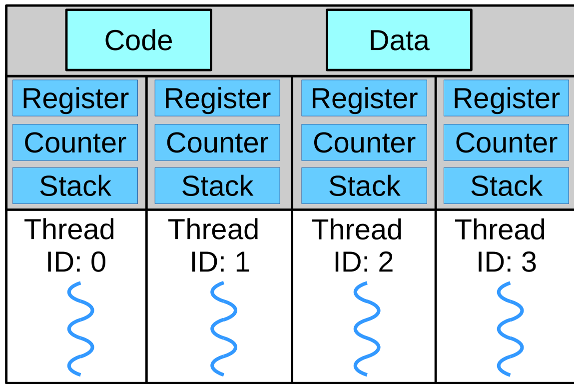

FIGURE 2. An illustration of a multi-threaded application.

spanning engineering, scientific, and graphic applications. Conversely, the PARSEC suite introduces a collection of emerging applications in recognition, mining, and synthesis (RMS) [50]. This suite also presents multi-threaded applications characteristic of commercial programs, including animation, media processing, enterprise servers, computer vision, and computational finance applications. Utilizing both SPLASH-2 and PARSEC offers diversity in aspects like working set size, cache miss rate, and instruction distribution [51].

A multi-threaded application encompasses multiple threads. Each thread is an independent task, and while all threads share a common data space, each possesses a unique thread ID, a register set, a stack, and a program counter [52]. This structure is visualized in Fig. 2. In this paper, the terms thread and task are used interchangeably. In this work, we utilize nine compute- and memory-intensive multi-threaded applications from the PARSEC and SPLASH-2 benchmark suites to assess the efficacy of our proposed work. These applications are *Blackscholes*, *Bodytrack*, *Cholesky*, *Dedup*, *FFT*, *Fluidanimate*, *Ocean*, *Radix*, and *Raytrace*. For more details about these applications' characteristics, please refer to Ref. [53].

C. PROBLEM FORMULATION

Consider a 3D many-core system, which consists of **C** cores and **M** memory channels, that runs multi-threaded applications. Given that only 50% of the cores and memory channels can be active at any given time due to the thermal constraints, the goal of our proposed technique is to minimize the total execution time E_t, which refers to the overall time taken for the execution of the multi-threaded applications in a 3D many-core system. Simultaneously, we aim to ensure that the temperatures of the cores and memory banks do not exceed a specified threshold temperature T_{th}. This goal can be expressed in mathematical terms as follows:

$$\text{Minimize } E_t \quad \text{s.t. } T_c, T_m < T_{th}, \\ \text{for all } c \in \{0, \cdots, \boldsymbol{C}-1\}, m \in \{0, \cdots, \boldsymbol{M}-1\} \quad (1)$$

Here, T_c represents the transient temperature of core c and T_m represents the transient temperature of memory channel m. This formulation takes into account the transient temperatures of both the cores and memory banks.

IV. PROPOSED 3D-DNaPE TECHNIQUE

This section outlines the methodology of our proposed 3D-DNaPE technique. The objective of our technique is to dynamically enhance the performance of thermally constrained 3D many-core systems while considering thermal constraints. Thus, it is crucial for our suggested technique to be computationally lightweight. Task migration and DVFS are suitable lightweight options for managing chip thermal conditions in real-time if used efficiently. Task migration can effectively lower the system temperature by shutting off hot cores and moving tasks to cooler ones. In contrast, DVFS can gradually reduce the system temperature by incrementally lowering the DVFS level of hot cores when there is insufficient thermal headroom for task migration.

The proposed 3D-DNaPE takes into account vertical heat transfer across all layers of the 3D stack. Therefore, the DTM techniques, namely task migration and DVFS, should consider the temperatures of both the core and memory bank layers in the 3D many-core system. Unlike DTaPO [17], which migrates tasks across all cores to maintain the checkerboard pattern, 3D-DNaPE uses a neighbor-aware pattern. It checks all the dark neighbor cores of the current core and selects the coldest one as the migration destination. Migrating tasks only to neighbor cores in thermally constrained 3D many-core systems has the advantage of reducing the search complexity and minimizing data transfer overhead.

To simplify the search for all surrounding neighbors, their indexes need to be found. To do that, first, the coordinates (x, y) of the current core on the NoC are calculated based on the current core's position using Eq. (2, 3).

$$x = \lfloor C_{index} \div w \rfloor, C_{index} = 0, \ldots, C-1 \quad (2)$$
$$y = C_{index} \,\%\, w, C_{index} = 0, \ldots, C-1 \quad (3)$$

where x is the row coordinate of the current core, y is the column coordinate of the current core, C_{index} is the current core index, and C is the total number of cores. After finding the current core coordinates, the surrounding neighbor core coordinates (n_x, n_y) are calculated based on the current core's coordinates. Finally, Eq. (4) is used to find the neighbor core index.

$$N_{index} = n_x \times w + n_y \quad (4)$$

where N_{index} is the index of the neighbor core, n_x is the row coordinate of the neighbor core, and n_y is the column coordinate of the neighbor core. Further details regarding the use of these equations can be found in Algorithm 1.

The proposed 3D-DNaPE uses task migration to transfer tasks to the coldest dark neighbor under two distinct conditions. The first condition is when the temperatures of both the coldest neighbor and its associated memory channel are lower than the predefined threshold temperature by a specific margin. The second condition is when the temperature of

the coldest neighbor and its associated memory channel are lower than the current core temperature by a specific margin. In scenarios that do not meet these conditions, 3D-DNaPE uses DVFS to reduce the frequency level of the current core.

3D-DNaPE uses heuristic algorithms to optimize performance while managing thermal conditions in a complex, dynamic 3D many-core system. The complexity and dynamism of the system justify the choice of a heuristic algorithm instead of finding an optimal solution. These algorithms aim to quickly find satisfactory solutions by leveraging problem-specific knowledge and simplifying assumptions. They are adept at providing practically good solutions, particularly in the dynamic environments typical of many-core systems. In these environments, cores experience heating and cooling phases, tasks exhibit varying arrival and execution patterns, and what may be an optimal configuration at one moment might lose its optimality shortly thereafter due to these dynamic changes. In essence, the deployment of heuristic algorithms in this context represents a prudent trade-off between optimality and computational efficiency, ensuring robustness amid uncertainty and dynamic change. The details of the proposed algorithms are explained in the following subsection.

A. PROPOSED ALGORITHMS

The proposed 3D-DNaPE first identifies the coldest destination core and then applies DTM techniques, specifically task migration and DVFS. Therefore, we have proposed two algorithms: Algorithm 1 (NaP) and Algorithm 2 (3D-DTM). All symbols used in these algorithms are defined in Table 1. Algorithm 1 is used to find the index of the neighbor that has the lowest temperature. This information is crucial for 3D-DNaPE to pattern the active and dark cores based on their neighbor temperatures. Algorithm 1 takes an index of current core C_{index}, a width of NoC w, a vector of all core temperatures $\boldsymbol{T_c}$, and a vector of all core statuses $\boldsymbol{S}$. It searches for the destination index D_{index} that has the lowest temperature. Initially, the active and dark cores are distributed evenly so that dark cores encircle each active core, as shown in Fig. 1.

To find the current core's coordinates on the NoC, Eq. (2, 3) are used (lines 1-2). A minimum temperature T_{min} is initialized to the temperature of the current core (line 3). All coordinates of the eight neighboring cores around the current core are stored in the $\boldsymbol{N}$ array (line 4). The algorithm then iterates over each neighbor in the $\boldsymbol{N}$ array. For each neighbor, it retrieves the row coordinate n_x and column coordinate n_y. If the neighbor coordinates are within the bounds of the NoC (i.e., $0 \leq n_x < w$, and $0 \leq n_y < w$), it calculates the neighbor index N_{index} using Eq. (4). If the temperature value at the neighbor index $T[N_{index}]$ is less than T_{min} and it is a *dark* core, it updates the destination index D_{index} with the neighbor index and updates T_{min} with the temperature neighbor. Finally, the algorithm returns the index of the core that has the minimum temperature among the neighbors, which is used by Algorithm 2.

TABLE 1. Definitions of symbols used in the proposed work.

Symbol	Definition
$CoreIndex$	current core index
w	NoC mesh width
N_{index}	Neighbor index
D_{index}	Destination index
A_c	All active cores set
a_i	Active core $i \in A_c$
D_c	All dark cores set
d_i	Dark core $i \in D_c$
T_{min}	A minimum temperature
N	Array of all neighboring core coordinates
T_c	Set of all core transient temperatures
T_{th}	Threshold temperature
n_x	Row coordinate of neighbor core
n_y	Column coordinate of neighbor core
T_m	Memory channels transient temperatures set
f	Frequency set for all cores
f_{th}	Threshold frequency
S	Set of core statuses (active/dark)
x	Row coordinate of the current core
y	Column coordinate of current core
α	Safe-margin value
δ	Frequency level step

Algorithm 2 utilizes task migration to move tasks from hot active cores to cooled dark cores, allowing the 3D many-core system to operate at high performance without exceeding thermal limits. If there is insufficient thermal headroom for task migration, the algorithm applies DVFS for more progressive thermal reduction. The algorithm takes several inputs, including all cores' transient temperatures, represented by $\mathbf{T_c}$; the transient temperature of all memory channels, represented by $\boldsymbol{T_m}$; an active cores' set, denoted as $\boldsymbol{A_c} = \{a_0, \ldots, a_{n-1}\}$; a dark cores' set, denoted as $\boldsymbol{D_c} = \{d_0, \ldots, d_{n-1}\}$; and a set of all active cores' frequency levels, denoted as $\boldsymbol{F} = \{f_0, \ldots, f_{n-1}\}$. Additionally, the algorithm reads threshold temperature T_{th}, threshold frequency f_{th}, a safe-margin value α, and a frequency level step δ from a configuration file.

At a predefined control interval, the algorithm checks the temperature of each active core and its associated memory channel (lines 1-3). If any of them exceed the threshold temperature, Algorithm 1 is utilized to determine the destination index of a candidate dark core with the lowest temperature among its neighboring dark cores (line 4).

If the temperature of the candidate dark core and its associated memory channel is lower than the threshold temperature by a sufficient margin α, the candidate dark core is activated, and the current core is deactivated (lines 5-8). In case the previous condition is not met, the algorithm proceeds to check if the temperature of the candidate dark core and its associated memory channel is lower than the threshold temperature by α. If this condition holds, the algorithm activates the dark core, reduces its frequency using DVFS, and deactivates the current core (lines 9-14).

Otherwise, if there is no thermal headroom available for the task migration, the algorithm uses DVFS to reduce the frequency of the current core by δ (line 17). In cases where

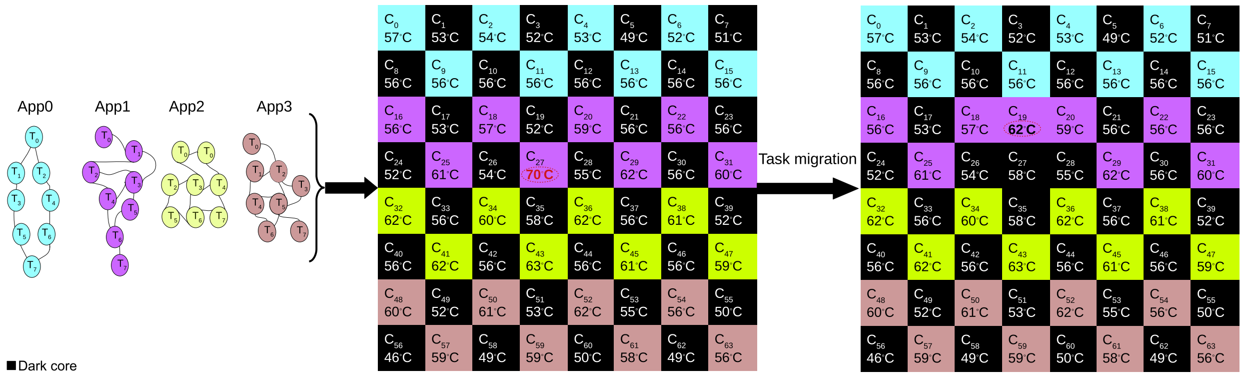

FIGURE 3. An illustrative example demonstrating the functionality of the proposed algorithms.

Algorithm 1 Neighbor-Aware Pattern Algorithm

Data: C_{index}, w, $\boldsymbol{T_c}$, $\boldsymbol{S}$
Output: D_{index}

1 Use Eq. (2, 3) to find (x, y);
2 $T_{min} \leftarrow T_c[C_{index}]$;
3 $N \leftarrow \begin{bmatrix} x-1 & y \\ x+1 & y \\ x & y-1 \\ x & y+1 \\ x-1 & y-1 \\ x-1 & y+1 \\ x+1 & y-1 \\ x+1 & y+1 \end{bmatrix}$;
4 **for** k=*0* **to** *7* **do**
5   $n_x \leftarrow neighbors[k][0]$;
6   $n_y \leftarrow neighbors[k][1]$;
7   **if** $(nx \geq 0$ **and** $nx < w)$ **and** $(ny \geq 0$ **and** $ny < w)$ **then**
8   Use Eq. (4) to find N_{index} ;
9   **if** $T_c[N_{index}] < T_{min} and S[N_{index}] == dark$ **then**
10       $T_{min} \leftarrow T[D_{index}]$;
11       $D_{index} \leftarrow N_{index}$;
12   **end**
13   **end**
14 **end**
15 **return** D_{index};

Algorithm 2 3D-DTM algorithm

Input: $\boldsymbol{A_c}$, $\boldsymbol{D_c}$, $\boldsymbol{T_m}$, $\boldsymbol{T_c}$, $\boldsymbol{f}$, T_{th}, F_{th}, α, and δ
Output: Updated $\boldsymbol{A_c}$, $\boldsymbol{D_c}$, and $\boldsymbol{f}$

1 **while** *true* **do**
2   **for** $\forall a_i \in Ac$ **do**
3   **if** $\mathrm{T_c}[a_i] > \mathrm{T}_{th} or \mathrm{T_m}[a_i] > \mathrm{T}_{th}$ **then**
4   Use Algorithm 1 to find the destination index (D_{index});
5   **if** $\mathrm{T_c}[D_{index}] < \mathrm{T}_{th} - \alpha and \mathrm{T_m}[D_{index}] < \mathrm{T}_{th} - \alpha$ **then**
6         $\mathrm{S}[D_{index}]$= active;
7   Move the task from $A_c[a_i]$ to $D_c[D_{index}]$;
8         $\mathrm{S}[a_i]$= dark;
9   **end**
10   **else if** $\mathrm{T_c}[d_i] < \mathrm{T_c}[a_i] - \alpha and \mathrm{T_m}[d_i] < \mathrm{T_m}[a_i] - \alpha$ **then**
11         $\mathrm{S}[D_{index}]$= active;
12   Reduce $f[D_{index}]$ by δ;
13   Move the task from $A_c[a_i]$ to $D_c[D_{index}]$;
14         $\mathrm{S}[a_i]$= dark;
15   **end**
16   **else**
17   Reduce $f[a_i]$ by δ;
18   **end**
19   **end**
20   **else if** $\mathrm{T_c}[a_i] < \mathrm{T}_{th} - \alpha$ *and* $f[a_i] < f_{th}$ **then**
21   Increase $f[a_i]$ by δ;
22   **end**
23   **end**
24 **end**

there are no thermal violations and the frequency is less than f_{th}, the algorithm increases the frequency of the current core by δ to improve system performance (lines 20-21).

In summary, the 3D-DNaPE algorithm periodically monitors the temperature of active cores and their associated memory channels, initiating appropriate actions based on the temperature readings of the cores and memory banks. Fig. 3 provides a visual representation of the proposed algorithms in action. For simplicity, while not losing generality, this illustration focuses solely on core temperatures. As depicted, the temperature of Core 27 (C_{27}) exceeds the threshold—set at 65°C for this example—prompting the NaP algorithm to find out the coldest core among the neighboring inactive ones, which include C_{19}, C_{26}, C_{28}, and C_{35}. In this scenario, Core 19 (C_{19}) is identified as the coldest. Subsequently, the

3D-DTM algorithm activates C_{19}, facilitates the migration of task from C_{27} to C_{19}, and then deactivates C_{27} to allow it to cool.

B. COMPLEXITY ANALYSIS

The time complexity of the NaP algorithm is constant time $\mathcal{O}(1)$ irrespective of the input size. It performs a fixed sequence of operations to evaluate the thermal conditions of neighboring cores and identify a suitable candidate for task migration. The fixed-size array of neighboring cores and the iteration through this array, along with other operations within the algorithm, all contribute to the constant time complexity. On the other hand, the 3D-DTM algorithm primarily hinges on the inner `for` loop that iterates through all active cores in the set $\boldsymbol{A_c}$, resulting in a linear time complexity of $\mathcal{O}(n)$ where n is the number of active cores. Within this loop, various operations are performed, including temperature checking and invocation of Algorithm 1, both of which operate in constant time $\mathcal{O}(1)$. The task migration steps and frequency scaling operations within this loop are also constant-time operations, thereby maintaining the overall linear time complexity of $\mathcal{O}(n)$.

Regarding space complexity, the NaP algorithm exhibits a constant space complexity $\mathcal{O}(1)$. It utilizes a fixed-size array to hold the coordinates of neighboring cores and a few other variables to perform its operations that do not scale with the input size, thereby rendering a constant space complexity. Conversely, the 3D-DTM algorithm's space complexity is linear $\mathcal{O}(n)$, primarily due to the data structures used to represent the active and dark cores, transient temperatures, and frequency levels. These data structures are likely to scale with the number of cores and memory channels in the system.

V. EXPERIMENTAL EVALUATION

Numerous experiments were carried out to assess the validity and efficacy of our proposed work. This section presents the details of the experimental setups, the obtained comparison results, and a thorough discussion and analysis of these outcomes.

A. EXPERIMENTAL SETUP

The proposed 3D-DNaPE was validated on a 3D many-core system comprising three layers: one core layer and two memory layers. The core layer contains 64 cores, which are evenly split into 32 active cores and 32 dark cores, all interconnected via an 8×8 mesh-based NoC. It's worth noting that despite their shared instruction set architecture (ISA), these cores operate at heterogeneous frequencies. Their maximum clock frequency reaches 4 GHz. Each core occupies a space of 8.70 mm^2, based on McPAT [54] modeling designed for the 22-nanometer technology node. Each core is equipped with a 32-kilobyte private L1 data cache, a 32-kilobyte private L1 instruction cache, and a 64-kilobyte private L2 cache. Additionally, an 8-megabyte S-NUCA cache is used as an LLC. The S-NUCA cache is shared among all cores, i.e., 128-kilobyte per core. As for the memory layers, they contain 128 memory banks, divided evenly into two layers with 64 memory banks each. In terms of memory channels, there are 64 in total, with each channel servicing two memory banks. Table 2 provides a summary of this system's settings.

TABLE 2. Summary of system settings.

Setting	Value
No. of cores	64
Max. frequency	4 GHz
L1-I	32 KB
L1-D	32 KB
L2	512 KB
LLC	128 KB/core
Technology	22-nm
Memory Size	8 GB
Memory layers	2 layers
Memory channels	64 channels
Memory banks	128 banks
Connection type	8×8 mesh NoC

Fig. 4 illustrates the experimental framework of this study. We utilized the state-of-the-art CoMeT simulator for 3D many-core systems [55]. CoMeT is a toolchain that integrates Sniper [56], McPAT [54], CACTI [57], and HotSpot [58]. Sniper is a high-performance parallel simulator designed for x86-64 architecture, capable of simulating multi or many cores efficiently. In the realm of integrated power, area, and timing modeling, McPAT has garnered significant adoption, as evidenced by recent utilization [59], [60]. Its prominence stems from its ability to furnish exhaustive low-level configuration insights for processors operating in the multi/many-core domain. CACTI represents an innovative architecture-level integrated framework for modeling power, area, and timing aspects of cutting-edge memory technologies, including 3D-stacked memories, as well as traditional 2D DRAM and caches. This framework significantly simplifies the integration process with architectural-level core performance simulators, facilitating comprehensive evaluations of novel memory technologies. HotSpot simulator is one of the most commonly used tools for thermal simulations. This simulator is based on the well-known stacked-layer packaging technology.

To enable the modeling of thermally constrained many-core system, some modifications were made to the Sniper simulator. Specifically, adjustments were made to the Sniper scheduler, allocating tasks exclusively to active cores through a core mask pattern. Additionally, modifications were carried out in McPAT, allowing it to quantify only the dark cores' static power to model the dark silicon state. Moreover, the wake-up latency, which is the time it takes to transition from a dark state to an active state for each task migration, was modeled by adding 200 μs, following Linux's intel_driver. These enhancements allowed us to effectively study the implications of thermally constrained 3D many-core system in our experiment.

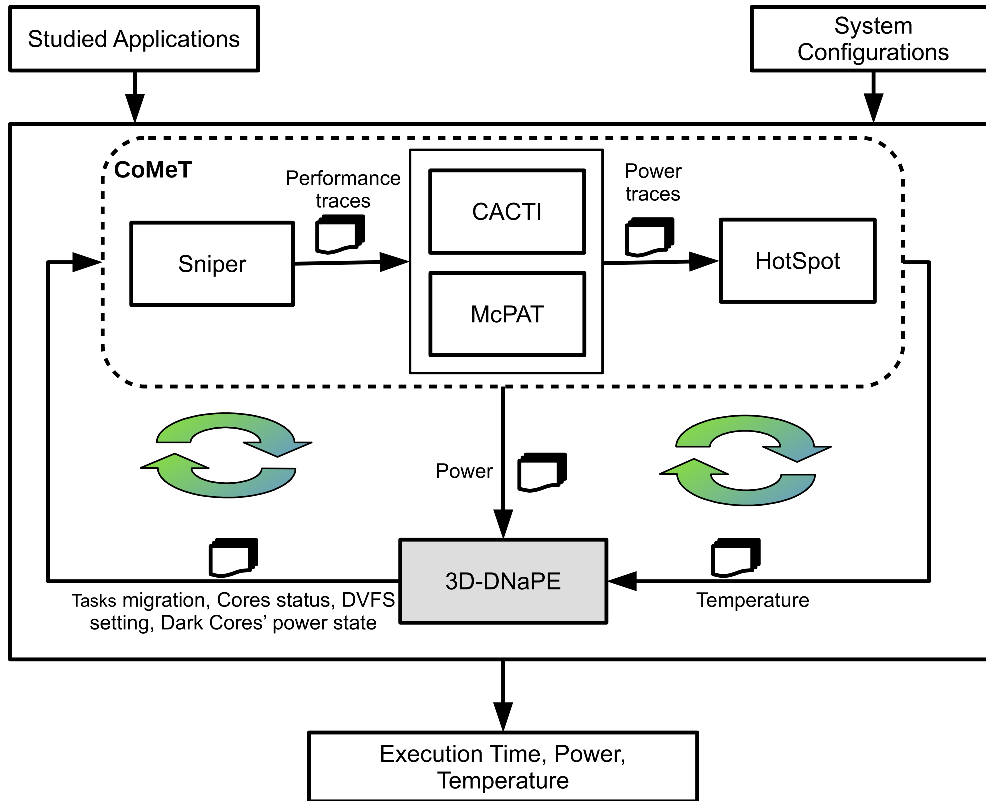

FIGURE 4. Experimental framework of the proposed work.

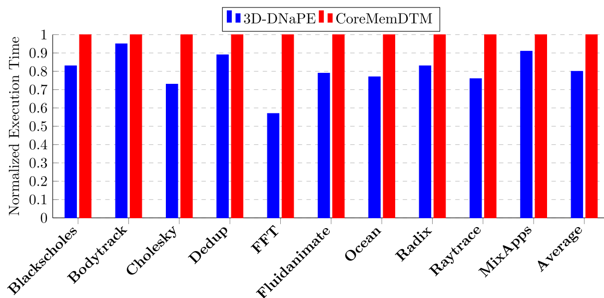

FIGURE 5. Normalized performance in terms of execution time.

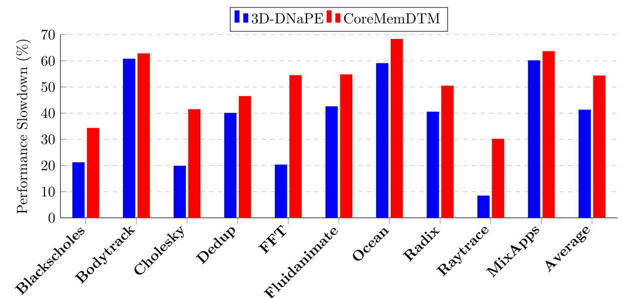

FIGURE 7. The performance slowdown percentage.

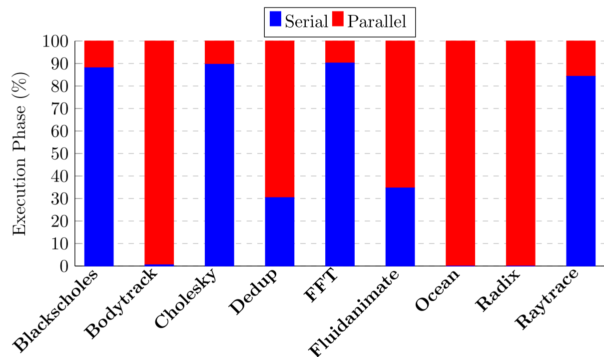

FIGURE 6. The percentage of serial and parallel execution phases within the studied applications.

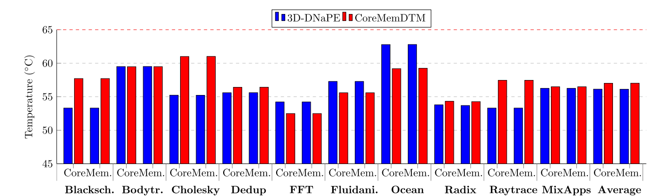

FIGURE 8. The average transient temperature of the studied applications.

The efficacy of the suggested technique was assessed through experimentation involving individual applications sourced from the PARSEC and SPLASH-2 benchmark suites. The studied applications are *Blackscholes*, *Bodytrack*, *Cholesky*, *Dedup*, *FFT*, *Fluidanimate*, *Ocean*, *Radix*, and *Raytrace*. Furthermore, a combination of compute- and memory-intensive applications is utilized to represent a diverse spectrum of computing demands, memory access patterns, and workload sizes. Within compute-intensive

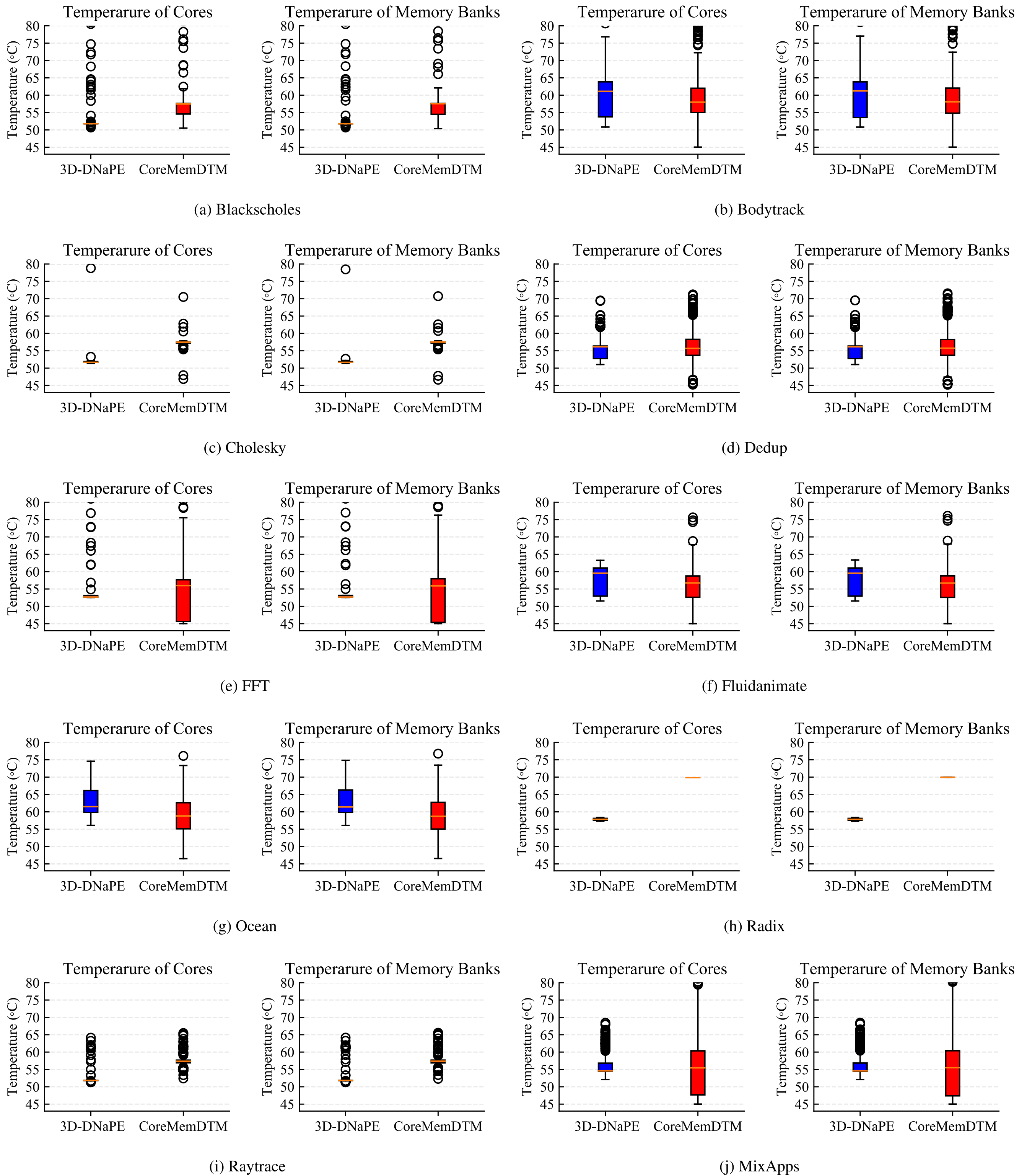

FIGURE 9. The temperature distributions of the 3D-DNaPE and CoreMemDTM techniques for the studied applications.

applications, tasks of elevated temperature can rapidly increase the temperature of cores. On the contrary, memory-intensive applications, characterized by a high volume of memory accesses, tend to drive up the temperature of the memory banks. The mixed applications, grouped as *MixApps*, are *Blackscholes*, *Bodytrack*, *Cholesky*, and *FFT*.

These applications are a mix of compute- and memory-intensive applications. Although our experiments mainly focus on specific applications from these benchmark suites, the underlying mechanisms and techniques are designed to be generalizable to large-scale real-world applications.

In our experimental setup, we configured the threshold frequency f_{th} and the frequency level δ to 200 MHz and 3800 MHz, respectively.The threshold temperature T_{th} was set to $65°C$. The safe margin value ε was set to 5% of the threshold temperature. The control period interval was set to 1 ms. The values of f_{th}, δ, and ε were empirically determined by conducting several experiments in which different values were tried. These values were chosen to ensure that the system does not frequently switch between active and dark cores. The value of T_{th} was selected to show the efficiency of the proposed work by considering the temperature characteristics of the studied applications as observed on the target platform. Ref. [17] provides additional details regarding the impact of the threshold temperature on the application of DTM techniques. The results presented in this paper are an average outcome of running the experiment ten times to mitigate the potential impact of random variations.

B. PERFORMANCE METRICS

The performance metrics used in our experiment validations are execution time E_t, performance slowdown $Perf_{slow}$, temperature, and power/energy. The execution time is the simulated execution time spent running a single simulation that starts at t_s until it finishes at t_f.

$$E_t = t_f - t_s \tag{5}$$

The performance slowdown $Perf_{slow}$ is the penalty for using DTM techniques. $Perf_{slow}$ is the difference between the execution time when a DTM technique is used $E_{t(DTM)}$ and when no DTM is used $E_{t(noDTM)}$.

$$Perf_{slow} = E_{t(DTM)} - E_{t(noDTM)} \tag{6}$$

The MIPS/W is the ratio between the instruction execution rate and the power consumption rate [61]. The MIPS/W metric is used to measure the energy efficiency of the proposed work. The instruction count I_{count} and execution time E_t are provided by Sniper. The power P is provided by McPAT and CACTI.

$$MIPS/W = \frac{I_{count}}{P \times t_E} \times 10^6 \tag{7}$$

C. COMPARATIVE RESULTS AND ANALYSIS

In this study, we evaluate the performance of the proposed 3D-DNaPE in comparison to the state-of-the-art CoreMemDTM [22], which takes into account both the cores and memory banks. Moreover, we compute the performance overhead for both our proposed 3D-DNaPE and CoreMemDTM by comparing their performance with the baseline, which does not incorporate any DTM technique.

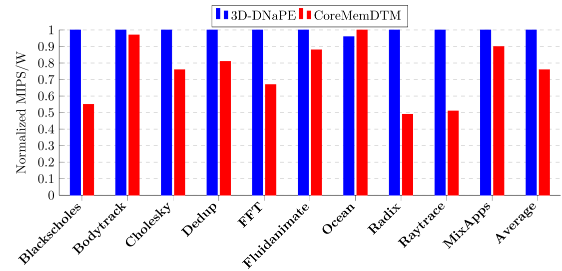

FIGURE 10. Normalized performance in terms of MIPS/W.

The performance results in terms of execution time were obtained by running the studied applications on the simulation setup outlined in the previous section and are shown in Fig. 5. This figure illustrates the normalized execution times of the studied applications when using our proposed 3D-DNaPE and CoreMemDTM. As can be seen, the proposed 3D-DNaPE outperforms CoreMemDTM's performance, demonstrating an improvement of up to 43% with an average enhancement of 20%. The extent of improvement is contingent upon the specific characteristics of each application [53]. Notably, applications with compute-intensive demands that benefit from heightened instruction-level parallelism capitalize on the increased frequencies offered by 3D-DNaPE, leading to substantial performance enhancements.

Conversely, CoreMemDTM activates all cores, presenting an advantage for applications that require substantial thread-level parallelism. Fig. 6 provides a depiction of the execution phases of the studied applications. Notably, applications like *Bodytrack*, *Ocean*, and *Radix* exhibit considerable parallel phases, benefiting from the large number of cores facilitated by CoreMemDTM. However, the disadvantage emerges when all cores are deactivated upon surpassing the maximum temperature threshold, resulting in overall system performance degradation.

To evaluate the performance slowdown caused by the DTM techniques, we use a baseline scenario. In this baseline scenario, the studied applications are executed without any thermal constraints, representing the maximum performance situation. Then, the execution time results of the proposed 3D-DNaPE and CoreMemDTM are compared against the baseline scenario according to Eq. (6). The percentage of performance slowdown for our proposed 3D-DNaPE and CoreMemDTM is shown in Fig. 7. Notably, our proposed 3D-DNaPE technique demonstrates a significantly lower performance slowdown, with reductions of up to 34% and an average of 13% when compared to CoreMemDTM. As highlighted in the previous discussion, the percentage of improvement varies depending on the unique characteristics of each application. Typically, applications with a high percentage of parallel phases tend to increase the temperature of the many-core system. Thus, this, in turn, prompts the more frequent activation of a DTM technique.

In terms of thermal management, both of our proposed 3D-DNaPE and CoreMemDTM manage to maintain the aver-

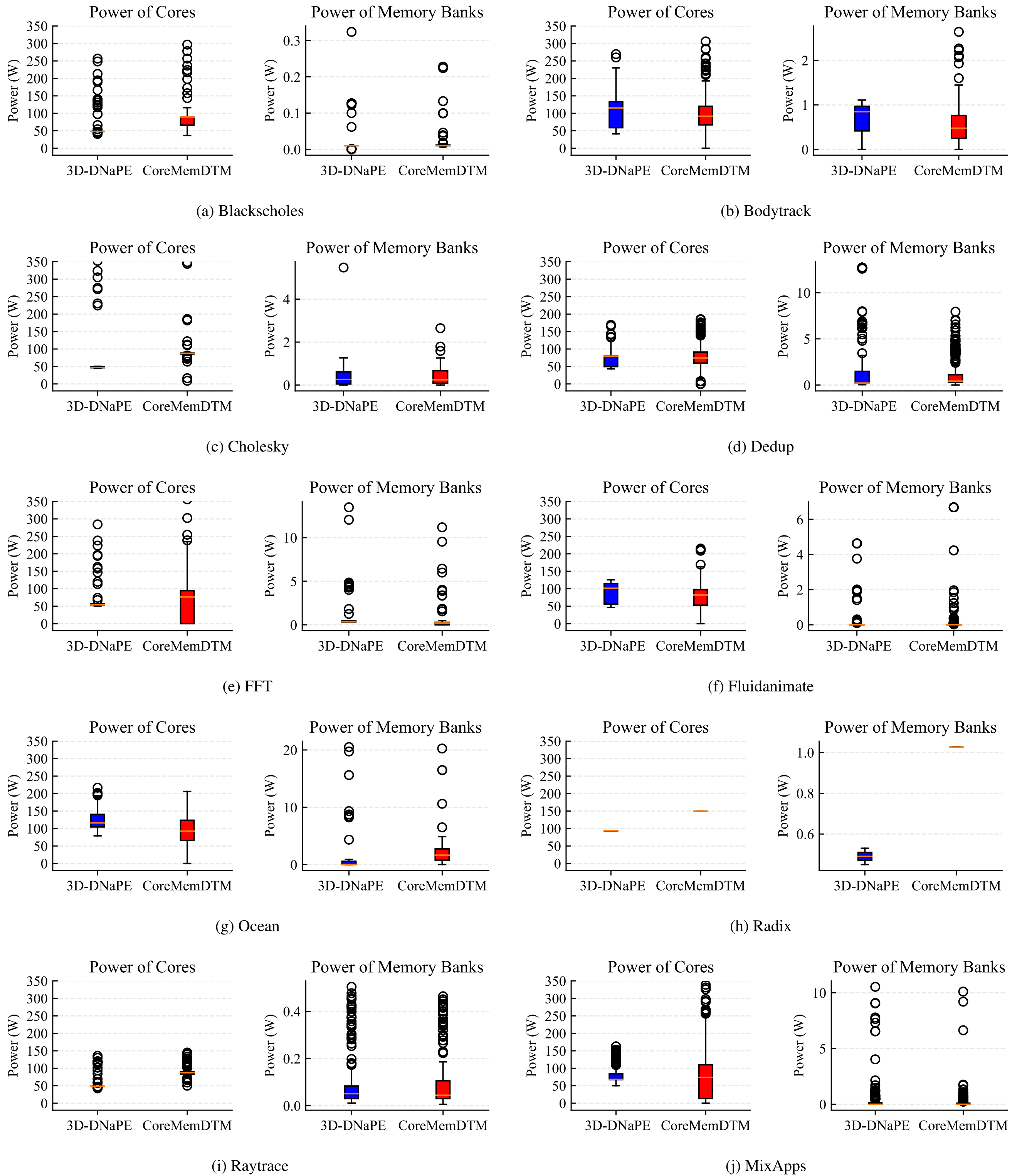

FIGURE 11. The statistical distribution of power consumption for the studied applications.

age system temperature lower than the predefined threshold. Fig. 8 shows the average transient temperature of running the studied applications under 65°C thermal constraint. As shown in our results, the proposed 3D-DNaPE outperforms CoreMemDTM in most of the studied applications by up to 9% and an average of 2%. In some applications like FFT, Fluidanimate, and Ocean, CoreMemDTM manages to lower the system temperature by 3%, 3%, 5.7%, respectively. This

temperature reduction is due to the deactivation of all cores. However, this leads to lower overall performance, as shown in Fig. 5.

The thermal statistical distribution of the proposed 3D-DNaPE and CoreMemDTM techniques can be observed from the box plots in Fig. 9. This figure shows the thermal behavior of cores and memory banks for the applications under study. Each subfigure represents the thermal distribution of the cores and memory banks for a specific application. For the majority of applications, the 3D-DNaPE technique exhibits up to 11°C less thermal variability compared to the CoreMemDTM technique. This observation implies that the 3D-DNaPE technique has a more consistent temperature profile for these applications. However, there are exceptions like *Bodytrack* and *Fluidanimate,* where CoreMemDTM demonstrates a reduced thermal variability by 3°C, 2°C, respectively. Also, applications like *FFT* and *Fluidanimate* show a slightly lower median temperature for CoreMemDTM. On the other hand, the range, as indicated by the whiskers, for the 3D-DNaPE technique is generally tighter in compute-intensive applications like *Blackscholes*, *Cholesky*, and *Radix*. This indicates that extreme temperatures are less frequent with our proposed 3D-DNaPE for these applications.

The energy efficiency is measured by MIPS/W according to Eq. (7). Fig. 10 shows the normalized performance in terms of MIPS/W of the proposed 3D-DNaPE and CoreMemDTM techniques. The proposed 3D-DNaPE shows higher energy efficiency for almost all the studied applications. The improvement is up to 51% and an average of 24% compared to CoreMemDTM. This is because the proposed 3D-DNaPE activates only half of the cores, which leads to less power consumption. However, for memory-intensive applications, CoreMemDTM matches the performance of 3D-DNaPE.

Similar to thermal statistical distribution analysis, we have also made box plots that show how cores and memory banks behave in terms of power for the applications being studied to look at the statistical distribution of power for the proposed 3D-DNaPE and CoreMemDTM techniques. Fig. 11 shows the statistical distribution of power consumption for the studied applications, where each subfigure represents the power consumption distribution of the cores and memory banks for a specific application. For compute-intensive applications, the 3D-DNaPE technique demonstrates up to 90 W less variability in power consumption compared to the CoreMemDTM technique. This suggests that the 3D-DNaPE technique has a more consistent power consumption profile for these applications. However, there are exceptions, such as *Fluidanimate*, where the variability for CoreMemDTM seems comparable to 3D-DNaPE. The range (as indicated by the whiskers) for the 3D-DNaPE technique is generally tighter in applications like *Blackscholes*, *Cholesky*, and *Radix*. This indicates that extreme power values (both high and low) are less frequent with 3D-DNaPE for these applications.

VI. CONCLUSION

This paper presents 3D-DNaPE, an innovative method to enhance the performance of thermally constrained 3D many-core systems. It selectively migrates tasks from hot cores to cooler dark cores, leveraging S-NUCA to reduce cache misses from thread migrations. Additionally, it adjusts DVFS to lower system temperature when no nearby cold dark cores are available. In a comprehensive assessment against CoreMemDTM, 3D-DNaPE consistently outperforms, improving execution time by up to 43% and reducing performance slowdown by up to 34%. It also excels in temperature regulation, with up to a 9% reduction in system temperature and a 51% enhancement in energy efficiency, achieved by activating only half of the cores. For future work, we plan to integrate a broader range of use cases representing different workload sizes. We also plan to do an in-depth analysis to enhance 3D-DNaPE's core activation strategy, potentially utilizing machine learning algorithms for application-specific core activation. Furthermore, we plan to propose a hybrid DTM technique, combining the strengths of 3D-DNaPE and CoreMemDTM to achieve superior performance across a broader range of applications.

ACKNOWLEDGMENT

The authors extend their appreciation to the Research Management Centre (RMC) at Universiti Teknologi Malaysia (UTM) for funding this research work under the Professional Development Research University grant (R.J130000.7113.06E45).

REFERENCES

[1] ITRS. *International Technology Roadmap for Semiconductors 2.0*. Accessed: Aug. 23, 2023. [Online]. Available: https://www.semiconductors.org/wp-content/uploads/2018/06/0_2015-ITRS-2.0-Executive-Report-1.pdf

[2] H. Esmaeilzadeh, E. Blem, R. S. Amant, K. Sankaralingam, and D. Burger, ''Dark silicon and the end of multicore scaling,'' in *Proc. 38th Annu. Int. Symp. Comput. Archit. (ISCA)*, San Jose, CA, USA, Jun. 2011, pp. 365–376.

[3] K. Cao, J. Zhou, T. Wei, M. Chen, S. Hu, and K. Li, ''A survey of optimization techniques for thermal-aware 3D processors,'' *J. Syst. Archit.*, vol. 97, pp. 397–415, Aug. 2019.

[4] D. Cuesta, J. L. Risco-Martin, J. L. Ayala, and J. Ignacio Hidalgo, ''3D thermal-aware floorplanner using a MOEA approximation,'' *Integration*, vol. 46, no. 1, pp. 10–21, Jan. 2013.

[5] L. Yavits, A. Morad, and R. Ginosar, ''The effect of temperature on Amdahl law in 3D multicore era,'' *IEEE Trans. Comput.*, vol. 65, no. 6, pp. 2010–2013, Jun. 2016.

[6] S. G. Kandlikar and A. Ganguly, ''Fundamentals of heat dissipation in 3D IC packaging,'' in *3D Microelectronic Packaging*. Berlin, Germany: Springer, 2017, pp. 245–260.

[7] S. S. Kumar, A. Zjajo, and R. van Leuken, ''Fighting dark silicon: Toward realizing efficient thermal-aware 3-D stacked multiprocessors,'' *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 25, no. 4, pp. 1549–1562, Apr. 2017.

[8] H. Khdr, S. Pagani, M. Shafique, and J. Henkel, ''Thermal constrained resource management for mixed ILP-TLP workloads in dark silicon chips,'' in *Proc. 52nd ACM/EDAC/IEEE Design Autom. Conf. (DAC)*, San Francisco, CA, USA, Jun. 2015, pp. 1–6.

[9] J. Wang, Z. Chen, J. Guo, Y. Li, and Z. Lu, ''ACO-based thermal-aware thread-to-core mapping for dark-silicon-constrained CMPs,'' *IEEE Trans. Electron Devices*, vol. 64, no. 3, pp. 930–937, Mar. 2017.

[10] A. Raghavan, Y. Luo, A. Chandawalla, M. Papaefthymiou, K. P. Pipe, T. F. Wenisch, and M. M. K. Martin, ''Computational sprinting,'' in *Proc. IEEE Int. Symp. High-Perform. Comput. Archit.*, New Orleans, LA, USA, Feb. 2012, pp. 1–12.

[11] A. Raghavan, L. Emurian, L. Shao, M. Papaefthymiou, K. P. Pipe, T. F. Wenisch, and M. M. K. Martin, ''Utilizing dark silicon to save energy with computational sprinting,'' *IEEE Micro*, vol. 33, no. 5, pp. 20–28, Sep. 2013.

[12] J. Zhan, Y. Xie, and G. Sun, ''NoC-sprinting: Interconnect for fine-grained sprinting in the dark silicon era,'' in *Proc. 51st ACM/EDAC/IEEE Design Autom. Conf. (DAC)*, Francisco, CA, USA, Jun. 2014, pp. 1–6.

[13] L. Shao, A. Raghavan, L. Emurian, M. C. Papaefthymiou, T. F. Wenisch, M. M. K. Martin, and K. P. Pipe, ''On-chip phase change heat sinks designed for computational sprinting,'' in *Proc. Semiconductor Thermal Meas. Manag. Symp. (SEMI-THERM)*, San Jose, CA, USA, Mar. 2014, pp. 29–34.

[14] A. Rezaei, D. Zhao, M. Daneshtalab, and H. Wu, ''Shift sprinting: Fine-grained temperature-aware NoC-based MCSoC architecture in dark silicon age,'' in *Proc. 53rd ACM/EDAC/IEEE Design Autom. Conf. (DAC)*, Austin, TX, USA, Jun. 2016, pp. 1–6.

[15] N. Morris, C. Stewart, L. Chen, R. Birke, and J. Kelley, ''Model-driven computational sprinting,'' in *Proc. 13th EuroSys Conf.*, Porto, Portugal, Apr. 2018, pp. 1–13.

[16] J. Wang, Z. Chen, S. Guo, Y. Li, and Z. Lu, ''Optimal sprinting pattern in thermal constrained CMPs,'' *IEEE Trans. Emerg. Topics Comput.*, vol. 9, no. 1, pp. 484–495, Jan. 2021.

[17] M. S. Mohammed, A. A. M. Al-Kubati, N. Paraman, A. A.-H. Ab Rahman, and M. N. Marsono, ''DTaPO: Dynamic thermal-aware performance optimization for dark silicon many-core systems,'' *Electronics*, vol. 9, no. 11, p. 1980, Nov. 2020.

[18] M. S. Mohammed, A. K. Al-Dhamari, A. A. A. Rahman, N. Paraman, A. A. M. Al-Kubati, and M. N. Marsono, ''Temperature-aware task scheduling for dark silicon many-core system-on-chip,'' in *Proc. 8th Int. Conf. Model. Simul. Appl. Optim. (ICMSAO)*, Manama, Bahrain, Apr. 2019, pp. 1–5.

[19] E. Ofori-Attah and M. O. Agyeman, ''AMA: An ageing task migration aware for high-performance computing,'' *J. Low Power Electron. Appl.*, vol. 13, no. 2, p. 36, May 2023.

[20] A. Asad, O. Ozturk, M. Fathy, and M. R. Jahed-Motlagh, ''Optimization-based power and thermal management for dark silicon aware 3D chip multiprocessors using heterogeneous cache hierarchy,'' *Microprocess. Microsyst.*, vol. 51, pp. 76–98, Jun. 2017.

[21] H. Wang, W. Li, W. Qi, D. Tang, L. Huang, and H. Tang, ''Runtime performance optimization of 3-D microprocessors in dark silicon,'' *IEEE Trans. Comput.*, vol. 70, no. 10, pp. 1539–1554, Oct. 2021.

[22] L. Siddhu, R. Kedia, and P. R. Panda, ''CoreMemDTM: Integrated processor core and 3D memory dynamic thermal management for improved performance,'' in *Proc. Design, Autom. Test Eur. Conf. Exhib. (DATE)*, Antwerp, Belgium, Mar. 2022, pp. 1377–1382.

[23] Y. Shen, S. Niknam, A. Pathania, and A. D. Pimentel, ''Thermal management for S-NUCA many-cores via synchronous thread rotations,'' in *Proc. Design, Autom. Test Eur. Conf. Exhib. (DATE)*, Antwerp, Belgium, Apr. 2023, pp. 1–6.

[24] C. Kim, D. Burger, and S. W. Keckler, ''An adaptive, non-uniform cache structure for wire-delay dominated on-chip caches,'' in *Proc. 10th Int. Conf. Architectural Support Program. Lang. Operating Syst.*, San Jose, CA, USA, Oct. 2002, pp. 211–222.

[25] C. Ramey, ''TILE-Gx100 ManyCore processor: Acceleration interfaces and architecture,'' in *Proc. IEEE Hot Chips 23 Symp. (HCS)*, Stanford, CA, USA, Aug. 2011, pp. 1–21.

[26] R. Devaraj, A. Sarkar, and S. Biswas, ''Supervisory control approach and its symbolic computation for power-aware RT scheduling,'' *IEEE Trans. Ind. Informat.*, vol. 15, no. 2, pp. 787–799, Feb. 2019.

[27] R. Devaraj and A. Sarkar, ''Resource-optimal fault-tolerant scheduler design for task graphs using supervisory control,'' *IEEE Trans. Ind. Informat.*, vol. 17, no. 11, pp. 7325–7337, Nov. 2021.

[28] A. Kanduri, M.-H. Haghbayan, A. M. Rahmani, M. Shafique, A. Jantsch, and P. Liljeberg, ''AdBoost: Thermal aware performance boosting through dark silicon patterning,'' *IEEE Trans. Comput.*, vol. 67, no. 8, pp. 1062–1077, Aug. 2018.

[29] B. Raghunathan and S. Garg, ''Job arrival rate aware scheduling for asymmetric multi-core servers in the dark silicon era,'' in *Proc. Int. Conf. Hardw./Softw. Codesign Syst. Synth.*, New Delhi, India, Oct. 2014, pp. 1–9.

[30] M. S. Mohammed, N. Paraman, A. A. A. Rahman, F. A. Ghaleb, A. Al-Dhamari, and M. N. Marsono, ''PEW: Prediction-based early dark cores wake-up using online ridge regression for many-core systems,'' *IEEE Access*, vol. 9, pp. 124087–124099, 2021.

[31] S. Pagani, H. Khdr, W. Munawar, J.-J. Chen, M. Shafique, M. Li, and J. Henkel, ''TSP: Thermal safe power-efficient power budgeting for many-core systems in dark silicon,'' in *Proc. Int. Conf. Hardw./Softw. Codesign Syst. Synth.*, New Delhi, India, Oct. 2014, pp. 1–10.

[32] H. Wang, D. Tang, M. Zhang, S. X.-D. Tan, C. Zhang, H. Tang, and Y. Yuan, ''GDP: A greedy based dynamic power budgeting method for multi/many-core systems in dark silicon,'' *IEEE Trans. Comput.*, vol. 68, no. 4, pp. 526–541, Apr. 2019.

[33] S. Niknam, A. Pathania, and A. D. Pimentel, ''T-TSP: Transient-temperature based safe power budgeting in multi-/many-core processors,'' *Power*, vol. 4, no. 6, p. 8, 2021.

[34] F. Hameed, M. A. A. Faruque, and J. Henkel, ''Dynamic thermal management in 3D multi-core architecture through run-time adaptation,'' in *Proc. Design, Autom. Test Eur.*, Grenoble, France, Mar. 2011, pp. 1–6.

[35] T.-H. Tsai and Y.-S. Chen, ''Thermal-aware real-time task scheduling for three-dimensional multicore chip,'' in *Proc. 27th Annu. ACM Symp. Appl. Comput.*, Trento, Italy, Mar. 2012, pp. 1618–1624.

[36] W.-K. Cheng and T.-W. Hsu, ''Thermal-aware task allocation, memory mapping, and task scheduling for 3D stacked memory and processor architecture,'' in *Proc. IEEE Tencon-Spring*. Sydney, NSW, Australia, Apr. 2013, pp. 95–98.

[37] D. Zhao, H. Homayoun, and A. V. Veidenbaum, ''Temperature aware thread migration in 3D architecture with stacked DRAM,'' in *Proc. Int. Symp. Quality Electron. Design (ISQED)*, Santa Clara, CA, USA, Mar. 2013, pp. 80–87.

[38] S. Aljeddani and F. Mohammadi, ''A novel migration technique to balance thermal distribution for future heterogeneous 3D chip multiprocessors,'' in *Proc. 8th Int. Conf. Inf. Sci. Technol. (ICIST)*, Cordoba, Spain, Jun. 2018, pp. 274–279.

[39] K. Kang, J. Kim, S. Yoo, and C.-M. Kyung, ''Runtime power management of 3-D multi-core architectures under peak power and temperature constraints,'' *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 30, no. 6, pp. 905–918, Jun. 2011.

[40] J. Meng, K. Kawakami, and A. K. Coskun, ''Optimizing energy efficiency of 3-D multicore systems with stacked DRAM under power and thermal constraints,'' in *Proc. DAC Design Autom. Conf.*, San Francisco, CA, USA, Jun. 2012, pp. 648–655.

[41] S. Lee, K. Kang, and C.-M. Kyung, ''Runtime thermal management for 3-D chip-multiprocessors with hybrid SRAM/MRAM l2 cache,'' *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 23, no. 3, pp. 520–533, Mar. 2015.

[42] S. Lee, K. Kang, J. Jung, and C.-M. Kyung, ''Hybrid L2 NUCA design and management considering data access latency, energy efficiency, and storage lifetime,'' *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 24, no. 10, pp. 3118–3131, Oct. 2016.

[43] Q. Zou, E. Kursun, and Y. Xie, ''Thermomechanical stress-aware management for 3-D IC designs,'' *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 25, no. 9, pp. 2678–2682, Sep. 2017.

[44] H. Wang, T. Xiao, D. Huang, L. Zhang, C. Zhang, H. Tang, and Y. Yuan, ''Runtime stress estimation for three-dimensional IC reliability management using artificial neural network,'' *ACM Trans. Design Autom. Electron. Syst.*, vol. 24, no. 6, pp. 1–29, Nov. 2019.

[45] H. Wang, D. Huang, R. Liu, C. Zhang, H. Tang, and Y. Yuan, ''STREAM: Stress and thermal aware reliability management for 3-D ICs,'' *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 38, no. 11, pp. 2058–2071, Nov. 2019.

[46] S. Spiesshoefer and L. Schaper, ''IC stacking technology using fine pitch, nanoscale through silicon vias,'' in *Proc. 53rd Electron. Compon. Technol. Conf.*, New Orleans, LA, USA, 2003, pp. 631–633.

[47] M. Shafique, S. Garg, J. Henkel, and D. Marculescu, ''The EDA challenges in the dark silicon era: Temperature, reliability, and variability perspectives,'' in *Proc. 51st Annu. Design Autom. Conf.*, Francisco, CA, USA, Jun. 2014, pp. 1–6.

[48] S. C. Woo, M. Ohara, E. Torrie, J. P. Singh, and A. Gupta, ''The SPLASH-2 programs: Characterization and methodological considerations,'' in *Proc. 22nd Annu. Int. Symp. Comput. Archit.*, Santa Margherita Ligure, Italy, 1995, pp. 24–36.

[49] C. Bienia, S. Kumar, J. P. Singh, and K. Li, ''The PARSEC benchmark suite: Characterization and architectural implications,'' in *Proc. Int. Conf. Parallel Architectures Compilation Techn. (PACT)*, Toronto, ON, Canada, Oct. 2008, pp. 72–81.

[50] P. Dubey, "Recognition, mining and synthesis moves computers to the era of tera," *Technol. Intel Mag.*, vol. 9, no. 2, pp. 1–10, 2005.

[51] C. Bienia, S. Kumar, and K. Li, "PARSEC vs. SPLASH-2: A quantitative comparison of two multithreaded benchmark suites on chip-multiprocessors," in *Proc. IEEE Int. Symp. Workload Characterization*, Seattle, WA, USA, Oct. 2008, pp. 47–56.

[52] A. Silberschatz, P. B. Galvin, and G. Gagne, *Operating System Concepts*, 9th ed. Hoboken, NJ, USA: Wiley, 2012.

[53] M. S. Mohammed and G. A. Abandah, "Communication characteristics of parallel shared-memory multicore applications," in *Proc. IEEE Jordan Conf. Appl. Electr. Eng. Comput. Technol. (AEECT)*, Amman, Jordan, Nov. 2015, pp. 1–6.

[54] S. Li, J. H. Ahn, R. D. Strong, J. B. Brockman, D. M. Tullsen, and N. P. Jouppi, "McPAT: An integrated power, area, and timing modeling framework for multicore and manycore architectures," in *Proc. 42nd Annu. IEEE/ACM Int. Symp. Microarchitecture (MICRO)*, New York, NY, USA, Dec. 2009, pp. 469–480.

[55] L. Siddhu, R. Kedia, S. Pandey, M. Rapp, A. Pathania, J. Henkel, and P. R. Panda, "CoMeT: An integrated interval thermal simulation toolchain for 2D, 2.5D, and 3D processor-memory systems," *ACM Trans. Archit. Code Optim.*, vol. 19, no. 3, pp. 1–25, Sep. 2022.

[56] W. Heirman, T. Carlson, and L. Eeckhout, "Sniper: Scalable and accurate parallel multi-core simulation," in *Proc. 8th Int. Summer School Adv. Comput. Archit. Compilation High-Performance Embedded Syst. (ACACES)*, 2012, pp. 91–94.

[57] N. Muralimanohar, R. Balasubramonian, and N. P. Jouppi, "Cacti 6.0: A tool to model large caches," *HP Laboratories*, vol. 27, p. 28, Apr. 2009.

[58] R. Zhang, M. R. Stan, and K. Skadron, "HotSpot 6.0: Validation, acceleration and extension," Dept. Comput. Sci., Univ. Virginia, Charlottesville, VA, USA, Tech. Rep. CS-2015-04, 2015.

[59] D. Gao, D. Reis, X. S. Hu, and C. Zhuo, "Eva-CiM: A system-level performance and energy evaluation framework for computing-in-memory architectures," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 39, no. 12, pp. 5011–5024, Dec. 2020.

[60] L. Shen, N. Wu, and G. Yan, "Fuzzy-based thermal management scheme for 3D chip multicores with stacked caches," *Electronics*, vol. 9, no. 2, p. 346, Feb. 2020.

[61] E. Grochowski and M. Annavaram, "Energy per instruction trends in Intel microprocessors," *Technol. Intel Mag.*, vol. 4, no. 3, pp. 1–8, 2006.

MOHAMMED SULTAN MOHAMMED (Member, IEEE) received the B.Sc. degree in computer engineering from Hodeidah University, Yemen, in 2005, the M.Sc. degree in computer engineering and networks from The University of Jordan, Jordan, in 2015, and the Ph.D. degree in electrical engineering (computer engineering) from Universiti Teknologi Malaysia (UTM), Malaysia, in 2022. He is currently a Researcher with Universiti Teknologi Malaysia, under the Postdoctoral Fellowship Scheme for the project "Thermal-Aware Performance Optimization of 3D Dark Silicon Many-Core Systems." His research interests include AI, computer architectures, many-core system-on-chip (MCSoC), network-on-chip (NoC), power and thermal management, and computer vision.

AHLAM AL-DHAMARI received the B.Sc. degree in computer engineering from Hodeidah University, Yemen, the M.Sc. degree in computer engineering and networks from the University of Jordan, Jordan, and the Ph.D. degree in electrical engineering from Universiti Teknologi Malaysia (UTM), Malaysia. She was a Postdoctoral Researcher under an international fellowship with Universiti Teknologi Malaysia. She is currently a Faculty Member with the Department of Computer Engineering, Faculty of Computer Science and Engineering, Hodeidah University. Her research interests include computer vision, machine learning, deep learning, image and video processing, computer architectures, big data analysis, and crowd analysis and management. She is a highly active reviewer for well-known international journals and conferences.

MOSAB HAMDAN (Senior Member, IEEE) received the B.Sc. degree in computer and electronic system engineering from the University of Science and Technology (UST), Sudan, in 2010, the M.Sc. degree in computer architecture and networking from the University of Khartoum (UofK), Sudan, in 2014, and the Ph.D. degree in electrical engineering (computer networking) from the Faculty of Engineering, School of Electrical Engineering, Universiti Teknologi Malaysia (UTM), Malaysia, in 2021. From 2010 to 2015, he was a Teaching Assistant and a Lecturer with the Department of Computer and Electronics System Engineering, Faculty of Engineering, University of Science and Technology (UST). He is currently a Researcher with the Interdisciplinary Research Center for Intelligent Secure Systems, King Fahd University of Petroleum and Minerals, Saudi Arabia. His current research interests include software-defined networking (SDN), load balancing, network traffic classification, the Internet of Things (IoT), cloud computing, network security, and future networks.

ABDUL-MALIK H. Y. SAAD (Senior Member, IEEE) was born in Jeddah, Saudi Arabia, in 1983. He received the B.Sc. degree (Hons.) in computer engineering from Hodeidah University, Yemen, in 2006, and the M.Sc. degree in electronic systems design engineering and the Ph.D. degree in digital systems from Universiti Sains Malaysia, in 2014 and 2018, respectively. His research interests include digital system design, image processing, and AI.

ANTAR S. H. ABDUL-QAWY received the bachelor's degree in computer engineering from Hodeidah University, Yemen, in 2005, the M.Tech. degree in computer science from the University of Hyderabad, India, in 2014, and the Ph.D. degree in electronics and communication engineering (Internet of Things) from Kakatiya University, India, in 2019. From 2005 to 2012, he was an Assistant Lecturer with the Department of Computer Engineering, Faculty of Computer Science and Engineering, Hodeidah University. He is currently a Senior Lecturer of information technology with the Department of Mathematics and Computer Science and the Dean of the Faculty of Science, Abdulrahman Al-Sumait University, Zanzibar, Tanzania. His research interests include the Internet of Things, wireless sensor networks, green IoT, and energy-efficient networks.

M. N. MARSONO received the B.Eng. degree in computer engineering and the M.Eng. degree in electrical engineering from Universiti Teknologi Malaysia, in 1999 and 2001, respectively, and the Ph.D. degree in ECE from the University of Victoria, in 2007. He is currently a Professor of electronic and computer engineering with the Faculty of Electrical Engineering, Universiti Teknologi Malaysia. His research interests include embedded systems, computer architecture, domain-specific reconfigurable computing, network processing architectures, network algorithmics, and software-defined networks.