Cover Letter

25-February-2023

To,
Prof. David Atienza
IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems (TCAD)

Sub: Request to consider our research manuscript for publication in TCAD

Dear Prof. Atienza,

We are glad to submit our research manuscript titled ”3D-TemPo: Optimizing 3D DRAM Performance Under Temperature and Power Constraints” for review and publication in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems (TCAD). This manuscript is authored by Shailja Pandey, Sayam Sethi, and Preeti Ranjan Panda of IIT Delhi, India.

In this manuscript, we address an important problem of performing efficient *dynamic power budgeting (DPB)* and *dynamic thermal management (DTM)* in 3D memories with a goal to maximize performance under a fixed power budget and a thermal constraint. Prior works have targeted power budgeting, alongside thermal management, primarily for CPU cores using approaches such as dynamic voltage and frequency scaling (DVFS), efficient task mapping, avoiding the clustering of active cores, learning and agent-based, transient temperature-aware budgeting, and dynamic programming. Prior art performing DTM in 3D DRAM has explored data migration based approaches, assuming only a thermal constraint for 3D DRAM and does not consider a memory power budget. For the first time, we propose an adjacency-aware dynamic power budgeting policy, *3D-TemPo*, which periodically suggests the ideal set of memory channels that should be activated under a given power budget and thermal constraint. Additionally, we consider a non-uniform traffic amongst memory channels, making power budgeting and thermal management decisions non-trivial. We perform an extensive experimentation using SPEC 2017 and PARSEC 2.1 benchmark suites to demonstrate the gains of our 3D-TemPo policy and observe speedups of 1x to 17.94x compared to baseline strategies.

We confirm that this manuscript is not under review anywhere else and comes with the approval of all authors for its submission to TCAD.

Thank You.

Yours Sincerely,

Shailja Pandey
(corresponding author)
Indian Institute of Technology Delhi, New Delhi, India.
Email: shailjapandey@cse.iitd.ac.in

3D-TemPo: Optimizing 3D DRAM Performance Under Temperature and Power Constraints

Shailja Pandey, Sayam Sethi, and Preeti Ranjan Panda

***Abstract*—3D DRAM provides a significant performance boost resulting from substantial memory bandwidth. However, the stacked memory architecture exhibits high power density, causing thermal hotspots. Further, systems under power constraints require careful planning for intelligent allocation of the available power to their various components. A straightforward dynamic power management policy of allocating more power to potentially high memory activity 3D DRAM ranks so as to maximize system performance causes a rise in the temperature of such ranks, making them susceptible to thermal stalls and shutdown by dynamic thermal management (DTM) strategies. A rise in rank temperature, in turn, increases the leakage power of memory ranks, affecting power budgeting decisions. Thus, a coordinated strategy for power budgeting and thermal management is needed. We propose an adjacency-aware dynamic power budgeting technique, *3D-TemPo*, which dynamically performs a reward-based power allocation to memory ranks, in order to maximize 3D DRAM performance under power and thermal constraints, and is sensitive to strong thermal correlations between vertically adjacent ranks. We evaluate *3D-TemPo* using *SPEC CPU2017* and *PARSEC 2.1* benchmark suites and observe speedups of 1x to 17.94x compared to baseline strategies.**

***Index Terms*—3D DRAM, Dynamic thermal management, Dynamic power budgeting.**

I. Introduction

Novel memory technologies such as 3D-stacked DRAMs [1], [2], offering substantial memory bandwidth (as high as 1 TBps), have been commercialised in an attempt to break the *memory wall*. High Bandwidth Memory (HBM) [1] is used in modern GPUs, and AI processors [3] to meet the bandwidth requirements of memory-intensive workloads involving huge data movement between processing cores and memory. Recent industry products such as Intel Xeon CPU Max Series [4] consist of x86 CPU cores and HBM memory to eliminate the memory bandwidth related bottlenecks. While 3D DRAM is a promising solution to bridge the speed gap between fast CPUs and slow memories, the vertically stacked integration of DRAM dies brings a new set of system design challenges. One such challenge is the poor heat dissipation capability of the 3D stacked architectures, causing excessive memory heating and eventually leading to throttling the memory and application slowdown. Further, heating in memory affects its data retention ability, and introduces additional overheads such as increased refresh rates [5], [6].

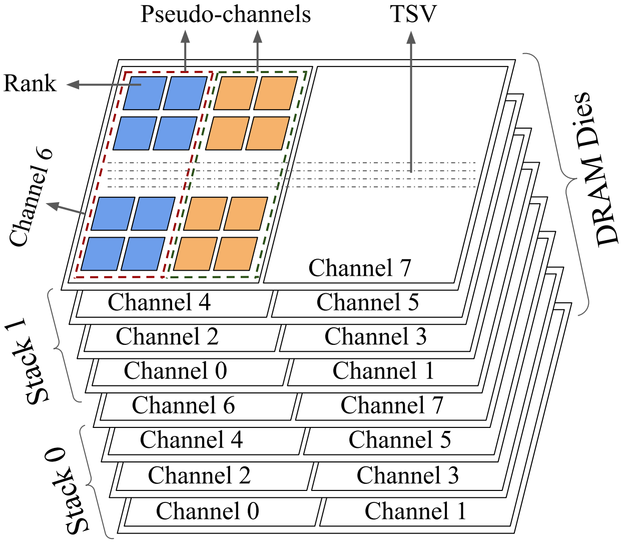


Fig. 1: High Bandwidth Memory (HBM2E) structure

Since the emergence of vertical integration technologies, power management and thermal challenges in memory have become first-order concerns. The vertical integration of DRAM dies with increased transistor densities results in higher thermal resistivities and power density, and poor heat dissipation capabilities. 3D stacking enables hundreds of banks/ranks and several independent channels per memory device such that each channel is connected to a separate memory controller [7]; however, the off-chip memory bandwidth is often limited due to its power budget [8]–[10]. Due to this, it is often not possible to activate and run all memory ranks simultaneously, similar to *dark silicon* in multi-core systems [11]–[14]. Systems running under a fixed power budget and a thermal constraint employ dynamic power budgeting (*DPB*) and dynamic thermal management (*DTM*) policies to ensure reliability [15]. Such policies that enable a subset of ranks and affect 3D DRAM performance are inter-dependent and cannot be performed in isolation. The static/leakage power of memory rises exponentially with increasing temperature [7], [16], motivating the need to perform thermal-aware power budgeting. Similarly, the temperature profile of memory ranks depends on the memory power budgeting decisions; the heating occurs from memory activity permitted by the power

This research was supported in part by Semiconductor Research Corporation (SRC).

Shailja Pandey is with the Department of Computer Science and Engineering, Indian Institute of Technology Delhi, New Delhi, India. Contact e-mail: shailjapandey@cse.iitd.ac.in

Sayam Sethi is with the Department of Computer Science and Engineering, Indian Institute of Technology Delhi, New Delhi, India. Contact e-mail: cs1190399@cse.iitd.ac.in

Preeti Ranjan Panda is with the Department of Computer Science and Engineering and Khosla School of IT at Indian Institute of Technology Delhi, New Delhi, India. Contact e-mail: panda@cse.iitd.ac.in

budgeting policy.

3D DRAM consists of several vertically stacked DRAM dies (Fig. 1) with varying power and thermal status. Each DRAM die consists of multiple physical channels, each channel composed of several ranks. The top dies, located closer to the heat sink, exhibit better heat dissipation; the bottom dies, located away from the heat sink, exhibit higher temperatures even when their memory activity (and therefore their dynamic power consumption) is low. Due to high temperatures in the bottom dies, the leakage power dissipation also continues to increase for the bottom dies. Thus, for the same amount of contribution to the system's *progress*, top dies experience low temperatures and dissipate less power than the bottom dies. The thermal characteristics of 3D DRAM make power budgeting a non-trivial task that requires run-time intervention to minimize the associated performance penalties. Further, 3D DRAM exhibits a strong thermal correlation between vertically adjacent DRAM ranks/channels compared to those within the same die [17], [18]. The thermal coupling in the vertical direction can be leveraged while allocating the power budget to channels, and motivates us to propose an adjacency-aware dynamic power budgeting strategy.

An important parameter affecting efficient dynamic power budgeting is the run time memory activity observed by different memory channels. As mentioned earlier, a 3D DRAM has multiple physical memory channels, each spanning over a disjoint portion of memory, and can regulate its corresponding accesses independently without interference from other channels. On one hand, such an architecture provides an opportunity to eliminate inter-application memory interference by mapping an application's data to a particular channel [19]. On the other hand, the variation in the memory traffic across channels and over time results in an uncertain thermal profile of channels. The application-to-channel mapping could potentially lead to a scenario where a highly utilized top channel is hotter than an under-utilized bottom channel, contrary to the typical thermal profile of 3D DRAM stack. This makes the power budgeting problem more interesting compared to when all applications uniformly access all the channels.

We make the following important observations in a thermally-constrained 3D DRAM system: (1) the physical location of a channel in the stacked architecture is crucial in determining the *system progress* obtained by enabling the channel, (2) allocating power budget to high memory activity channels does not always guarantee optimal performance as they are more prone to heating and thermal shutdown (due to high dynamic power dissipation), thereby stalling the cores, and (3) enabling vertically adjacent high memory activity channels could impede performance, as vertically aligned thermal hotspots drastically reduce 3D DRAM's cooling efficiency.

In this work, we make the following specific contributions:

1) We propose an adjacency-aware dynamic power budgeting policy, *3D-TemPo*, which periodically suggests the ideal set of channels that should be enabled under a given power budget and thermal constraint. This is the first work, to the best of our knowledge, towards a coordinated thermal and power management for 3D DRAM.
2) We consider a non-uniform traffic amongst memory channels, which results in a varying thermal profile across the 3D stack, making power budgeting decisions non-trivial.

II. Related Work

With increasing power density in novel memory technologies, power budgeting and thermal analysis of 3D architectures becomes an important concern. Modern data hungry workloads perform concurrent computations, requiring enormous amount of data movement between core and memory. The decision to turn on a portion of memory by judiciously utilizing the available power budget and without violating thermal constraints is non-trivial. Several run-time optimizations can be deployed to efficiently utilize the system power budget, enhancing performance [20]–[23]. Previous researches have targeted dynamic power budgeting on many/multi-core systems using different optimization metrics and design constraints. Prior works have proposed solutions to efficient power budgeting and thermal management primarily for homogeneous and heterogeneous many/multi-core systems using task migration and dynamic voltage and frequency scaling (DVFS) [24], thermal-aware application-to-core mapping [23], workload's power profile [25], QoS-aware frequency throttling [26], temperature prediction [27], [28], avoiding the clustering of active cores [29], learning and agent-based [30]–[32], transient temperature aware budgeting [33], [34], integer linear program (ILP) [35], and dynamic programming approaches [36]. Integrated CPU-GPU thermal and power management using DVFS [37], [38], and maximizing achievable frame rate [39] under fixed power budget has also been explored.

Thermal-aware design of 3D-stacked architectures has been gaining traction due to the evident thermal challenges [40]–[42]. For power and thermal management in 3D multi-core architecture (multiple layers of cores stacked together), Coskun et al. [43] proposed job scheduling and DVFS, Zhu et al. [44] leveraged the heterogeneous thermal characteristics of cores in different layers, and Meng et al. [45] used DVFS on cores to optimize energy efficiency, assuming uniform traffic across all DRAM banks. Prior works [7], [19], [46] have proposed application-aware memory channel partitioning, mapping cores to memory channels based on the applications running on the cores, in order to reduce inter-application interference. Such a mapping, which leads to non-uniform traffic across channels, is particularly advantageous for 3D DRAM that has several independent memory channels, each housing a disjoint portion of physical memory. In this work, we assume that different subsets of cores are mapped to different memory channels, generating non-uniform traffic across different 3D DRAM banks depending on the workload's run-time behavior.

Prior works have targeted power budgeting, alongside thermal management, primarily for CPU cores. Towards thermal management in 3D DRAM, FastCool [16] is a channel turn ON/OFF strategy that migrates data to a backup 2D DRAM upon thermal emergencies in 3D DRAM. NeuroMap [47] proposes an application-aware task mapping for deep neural networks and uses DRAM low-power states mechanism for DTM in 3D DRAM. However, both *FastCool* and *NeuroMap* assume only a thermal constraint for 3D DRAM and does not consider memory power budget. Addressing thermal issues in 3D DRAM under a given power budget requires joint evaluation of memory power, performance, and temperature [45]. Therefore, we focus on maximizing performance of a system with general-purpose cores and 3D DRAM through intelligent power budget allocation to memory channels such that all channels always operate within a safe thermal limit.

III. 3D DRAM Power Budgeting and Thermal Management

A. Memory Architecture

A 3D DRAM consists of multiple stacks, with each stack comprising several DRAM dies stacked vertically, connected using fast through-silicon via (TSV). A portion of the DRAM die from each stack is connected to the same independent physical channel, each channel consisting of two pseudo-channels with multiple ranks, which in turn contain banks that are organised into rows and columns, similar to a conventional 2D DRAM. Figure 1 shows the structure of the popular HBM2E with 2 stacks (0 and 1), 4 DRAM dies and 8 channels per stack, 2 pseudo-channels per channel, and 8 ranks per pseudo-channel. Each channel (0-7) services a total of 16 banks from two DRAM dies located in Stacks 0 and 1. A subset of CPU cores is mapped to each memory channel, restricting the memory traffic generated by each core to its corresponding channel only.

Modern DRAMs support multiple power states that are usually controlled at a rank level. Different power states differ in their power consumption with *read/write* state consuming maximum power (dynamic and leakage). The *standby* state, which does not permit memory accesses, consumes a fraction of the leakage power, required only to retain the data. In the *active* state, the memory consumes higher leakage power, and is ready for accesses. The power state transition of ranks requires a minimal overhead (typically of the order of a few ns to a few μs) [48]. Given a power budget, our dynamic power budgeting policy allocates power at the level of a memory channel. A channel is activated if all of its constituent ranks are either in *active* or *read/write* state. Similarly, to deactivate a channel, the budgeting technique sets all its ranks to *standby* state.

B. Motivation

We highlight the importance of performing a *coordinated* dynamic power budgeting (DPB) [45] and dynamic thermal management (DTM) in 3D DRAM, and discuss the potential performance, power, and thermal issues arising from executing the two policies in isolation.

- A dynamic power power budgeting (DPB) [49], [50] policy is responsible for opportunistically allocating the fixed memory power budget (P_b) to a subset of ranks/channels of 3D DRAM, sending other ranks to *standby* state.
- A dynamic thermal management (DTM) [7], [16], [51], [52] policy ensures that all the ranks of 3D DRAM always operate under the permissible thermal limit (T_{crit}), throttling the heated portions of memory until they cool down.

Figure 2 shows the thermal state of 3D DRAM, with applications running on cores mapped to corresponding memory channels, at time instances t, t', and t''. The thermal state of a channel is the result of: (1) memory activity initiated towards that channel from the cores, and (2) the physical location of the channel in the 3D stack. We show the decreasing order of channel productivity at different time instances, where channel productivity indicates the progress made by the applications upon activating the corresponding channel. Assuming a 50% memory power budget that permits activation of 4 channels simultaneously, an independently running DPB policy, unaware of channel temperatures, suggests allocating power to the most productive channels in that order. Similarly, an independently running DTM policy suggests sending channels approaching the thermal limit T_{crit} to *standby* state, temporarily stalling the memory accesses to those channels. We observe the following at different time instances, as illustrated in Figure 2:

1) At time t with channels 0 and 1 approaching the thermal limit T_{crit}:
   - *DPB* activates channels 0, 1, 2, and 3 based on the channel productivity.
   - *DTM* sets channels 0 and 1 to *standby* state.
   - Issue: *DTM* and *DPB* make *contradictory* decisions regarding the power states of channels 0 and 1.
2) At time t' with channels 2 and 3 approaching T_{crit}:
   - *DPB* does not assign power to channels 0 and 1 due to their not being among the top four productive channels.
   - *DTM* eventually sets channels 2 and 3 to *standby* state.
   - Issue: A temperature-aware *DPB*, instead of activating channels 2 and 3, would have rather activated less productive channels 0 and 1 that have been sufficiently cooled down as a result of the DTM decisions in the previous interval, preventing *over-compensation* in the system.
3) At time t'' with channels 0, 1, 2, and 3 approaching T_{crit}:
   - *DPB* suggests *standby* state for the least productive channels 4, 5, 6, and 7.
   - *DTM* sends heated channels 0, 1, 2, and 3 to *standby*.
   - Issue: The *complete* 3D DRAM becomes unavailable, unnecessarily stalling all the cores.

Thus, performing power and thermal management in isolation could potentially lead to following issues: (1) increased performance penalty from unnecessary CPU stalling, (2) inefficient utilization of the power budget due to DTM-induced throttling of the channels activated by *DPB*, and (3) increase in peak memory temperature due to *DPB* activating hotter channels. Based on the aforementioned scenarios, we conclude

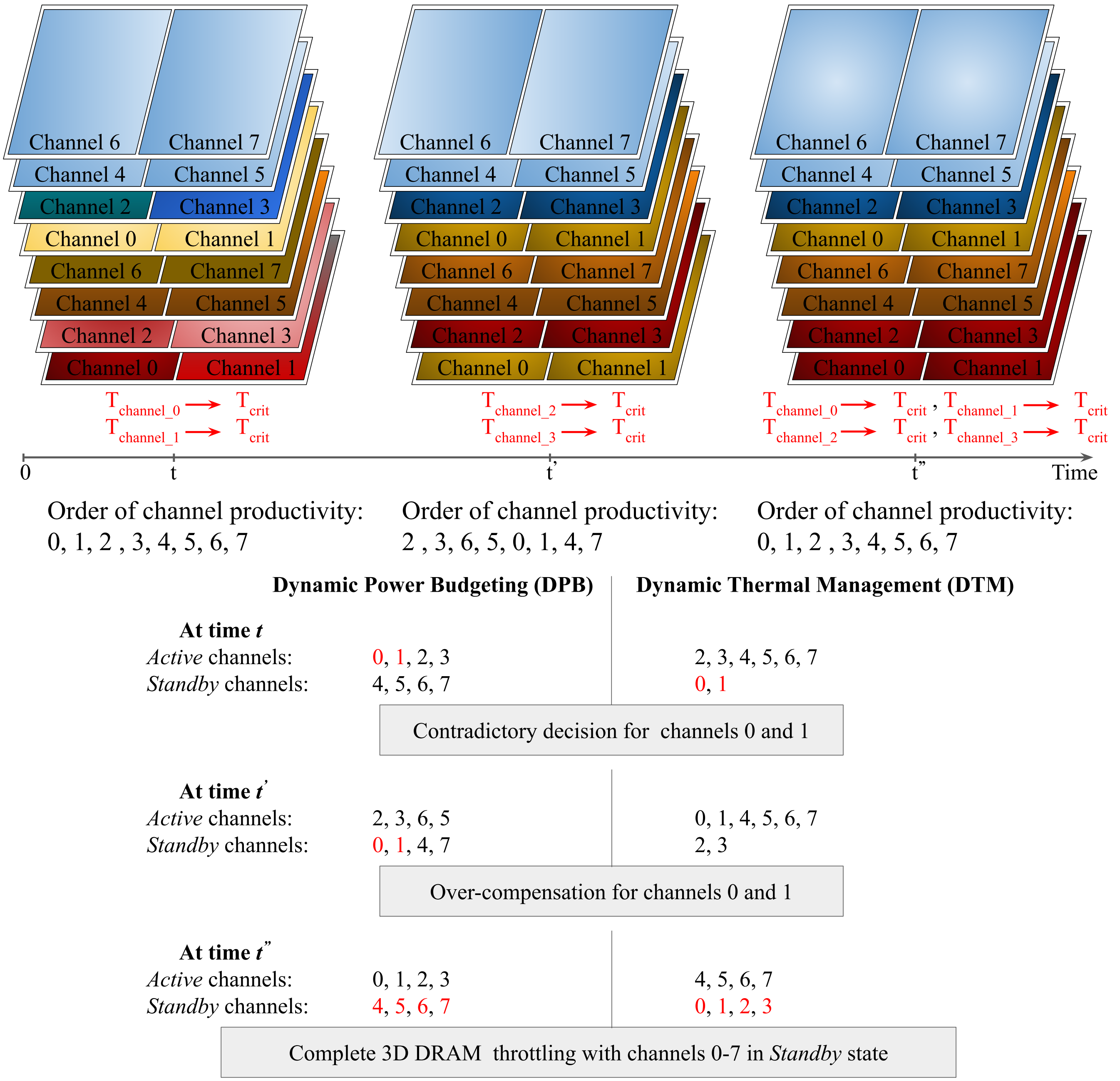

Channel 6
Channel 7
Channel 4
Channel 5
Channel 2
Channel 3
Channel 0
Channel 1
Channel 6
Channel 7
Channel 4
Channel 5
Channel 2
Channel 3
Channel 0
Channel 1
$T_{channel\_0} \rightarrow T_{crit}$
$T_{channel\_1} \rightarrow T_{crit}$
$T_{channel\_2} \rightarrow T_{crit}$
$T_{channel\_3} \rightarrow T_{crit}$
$T_{channel\_0} \rightarrow T_{crit}$, $T_{channel\_1} \rightarrow T_{crit}$
$T_{channel\_2} \rightarrow T_{crit}$, $T_{channel\_3} \rightarrow T_{crit}$
0
t
t'
t"
Time
Order of channel productivity: 0, 1, 2 , 3, 4, 5, 6, 7
Order of channel productivity: 2 , 3, 6, 5, 0, 1, 4, 7
Order of channel productivity: 0, 1, 2 , 3, 4, 5, 6, 7
Dynamic Power Budgeting (DPB)
Dynamic Thermal Management (DTM)
At time t
Active channels: 0, 1, 2, 3
Standby channels: 4, 5, 6, 7
2, 3, 4, 5, 6, 7
0, 1
Contradictory decision for channels 0 and 1
At time t'
Active channels: 2, 3, 6, 5
Standby channels: 0, 1, 4, 7
0, 1, 4, 5, 6, 7
2, 3
Over-compensation for channels 0 and 1
At time t"
Active channels: 0, 1, 2, 3
Standby channels: 4, 5, 6, 7
4, 5, 6, 7
0, 1, 2, 3
Complete 3D DRAM throttling with channels 0-7 in Standby state

Fig. 2: Dynamic power budgeting (DPB) and dynamic thermal management (DTM) working in isolation causes contradictory decisions, over-compensation, and complete memory throttling. The colors represent the channel temperatures – (1) shades of *red*: hotter channels, (2) shades of *yellow*: moderate temperatures, and (3) shades of *blue*: cooler channels.

that it is important to perform *DPB* and *DTM* in coordination, and propose a thermal-aware dynamic power budgeting policy for 3D DRAM while running under a fixed power budget, and a thermal constraint.

C. Problem Definition

The 3D DRAM thermal-constrained memory power budgeting (MPB) problem is formulated as follows. Given:

1) a multi-core processor with p processing cores,
2) a 3D DRAM memory with c channels, each channel mapped to p/c cores,
3) a memory power budget P_b, and
4) a temperature constraint T_{crit}

we aim to minimize the workload's total execution time, subject to:

$$T_{\mathrm{i}} \leq T_{\mathrm{crit}}, i \in \{0, 1, ..., (c-1)\}, and \\ \sum_{i \in \{0,1,...,(c-1)\}} P_i \leq P_b \tag{1}$$

where T_i and P_i denote the temperature and power consumption of channel i. The system execution is divided into intervals, and the MPB problem is defined in terms of the channel's power consumption and progress (made by the applications mapped to the channel) in the last interval the channel was active. The solution is used to define the memory configuration (power states of channels) for the next interval.

The MPB problem can be shown to be NP-Hard using a reduction from the well-known NP-Complete problem, *Knapsack*. Consider an instance of the Knapsack problem with n objects, with object i characterized by weight w_i and value v_i, and a maximum weight budget W. The problem is to choose a subset $S \subseteq \{1, ..., n\}$ such that $\sum_{i \in S} w_i \leq W$ and $\sum_{i \in S} v_i$ is maximized. We construct an MPB instance with channel i corresponding to object i, *progress* (measured as instructions-per-cycle (IPC) of the cores mapped to this channel in the last interval when the channel was active) $q_i = v_i$, power consumption P_i of channel i in the last active interval $= w_i$, power budget $P_b = W$, $T_{\text{crit}} = \infty$, and the subset of channels activated for the next interval corresponding to the selected subset S. Since a large T_{crit} value makes the temperature correlation effects across channels irrelevant, an efficient solution to the MPB problem directly solves the Knapsack problem due to the one-to-one correspondences, making MPB an NP-Hard problem.

D. Preliminary Power Budgeting Policies

Assuming a 3D DRAM with 8 channels and defining an interval (or epoch) as the duration between two consecutive invocations of a policy, we discuss the working of simplistic dynamic power budgeting policies below. We assume a 50% memory power budget that allows simultaneous activations of four memory channels in our illustrations of simplistic budgeting policies.

1) Round Robin Policy: Figure 3 illustrates the working of the round robin power budgeting policy. The round robin policy activates Channels 0 to 3 in the even intervals, and Channels 4 to 7 in the odd intervals of workload execution, with an interval (or epoch) consisting of a fixed time duration. This policy attempts to ensure fairness amongst all the memory channels and therefore amongst cores mapped to the channels. However, such a policy treats all the cores uniformly and often leads to sub-optimal performance as the cores running *compute-heavy* tasks, generating memory accesses at a much lower rate, undergo higher turnaround times. Furthermore, as stated earlier, 3D DRAM exhibits a strong thermal correlation between vertically adjacent channels/ ranks. Due to this, the round robin based activation order leads to frequent thermal stalls, causing performance penalties.

2) Alternation Policy: Figure 4 shows the working of the *alternation* policy that attempts to eliminate vertically aligned thermal hotspots in 3D DRAM. The policy allocates power to channels in alternate DRAM dies in both even and odd intervals. While this helps to improve DRAM cooling and reduces thermal stall durations, the higher turnaround times of *compute-heavy* applications still remains unresolved, affecting system performance. The policy also ensures fairness amongst cores; however, it often leads to sub-optimal performance as the memory activity rate in channels is not taken into consideration, leading to uniform treatment of highly accessed and less accessed channels. Wang et. al [29] proposes a similar idea for power budgeting in many-core systems which eliminates the clustering of active cores, reducing thermal hotspots in the close vicinity. The *Alternation* policy is inspired from the ideas presented in [29], and adapted for 3D DRAM.

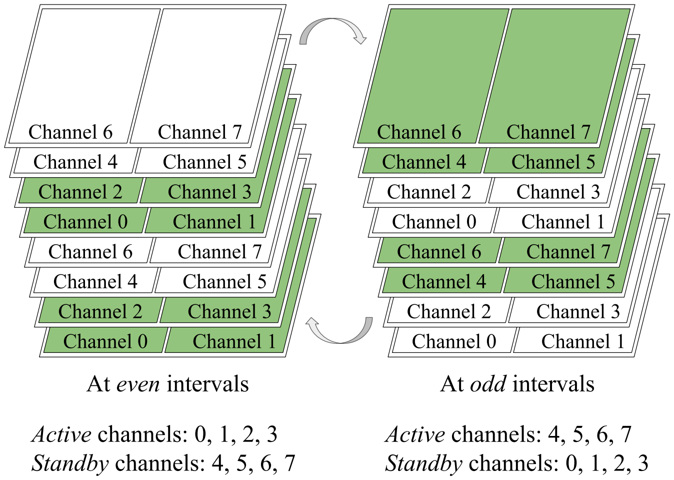


Fig. 3: Working of *RoundRobin* policy.

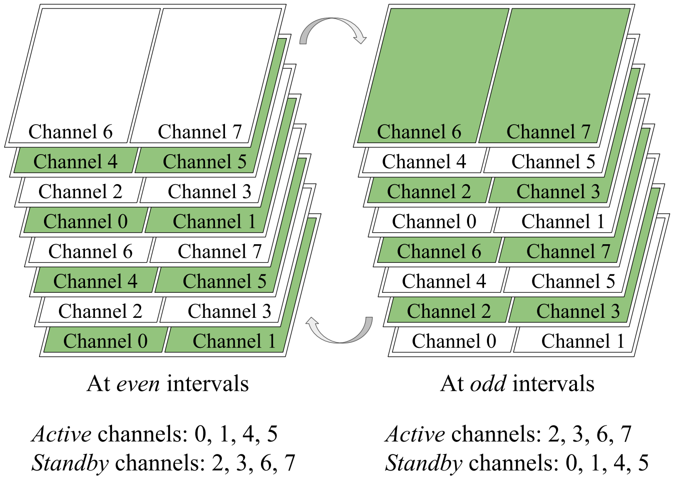


Fig. 4: Working of *Alternation* policy.

3) Most Frequently Used (MFU) Policy: The memory activity rate across different channels is an important indicator that determines 3D DRAM utilization. Figure 5 shows the working of the *most frequently used* (MFU) policy that maintains an MFU queue of channels for its operation. It allocates the available power budget to the most frequently used channels of the previous interval, leading to high memory throughput. For example, at time t, channels 0, 5, 3, and 1 at the head of MFU queue are activated. Similarly, at time t', channels 2, 1, 7, and 5 are activated. However, for thermally-constrained systems, a sharp rise in frequently used channels' temperatures is observed due to continuous activation. This, in turn, increases the number of thermal stalls due to high dynamic and leakage power dissipation. At each time interval, the channels at the head of the MFU queue are activated while the power budget permits. The policy ignores the accumulation of vertically aligned thermal hotspots and severely stalls the

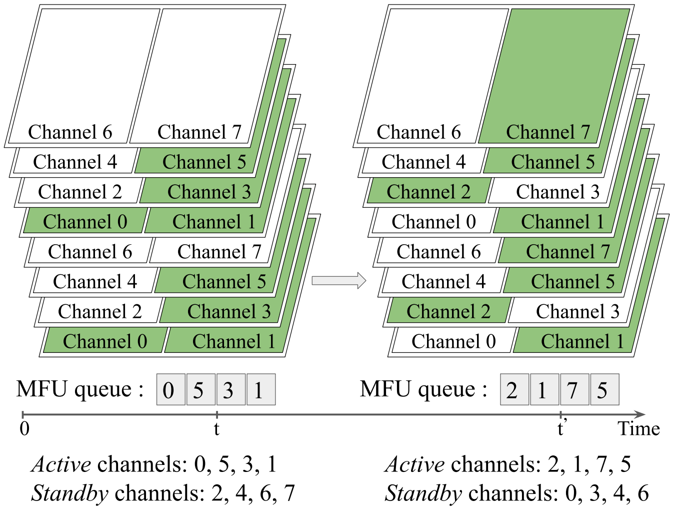


Fig. 5: Working of *MostFrequentlyUsed (MFU)* policy.

compute-heavy cores as the channels mapped to them are least preferred for activation. It only considers optimizing the immediate memory throughput overlooking the associated thermal impact.

4) Data Migration-based Policy: Prior work [16] has explored dynamic thermal management (DTM) approaches involving data migration to a backup memory upon heating in 3D DRAM in order to facilitate memory cooling. FastCool [16] turns off the heated memory channels and migrates the corresponding data to the backup 2D DRAM, incurring significant performance and energy overheads. As FastCool does not assume a memory power budget, we augment a dynamic power budgeting logic in the default *FastCool* algorithm to understand the feasibility of migration-based policies for thermal and power management. The modified algorithm prioritizes the channels that are activated at the beginning of each interval by the default *FastCool* policy, and allocates the power budget to these channels, migrating data from the other low-priority channels and turning them off. We discuss the performance implications of the data migration-based power budgeting in Section V-B.

IV. The 3D-TempPo Approach

In this section, we present our proposed *3D-TempPo* policy that jointly considers: (1) a memory channel's activity, (2) the *core progress* obtained through activating that channel, and (3) its physical location while allocating the power budget. We discuss the *reward* computation strategy, the main ideas of *3D-TempPo*, the DRAM low-power states based DTM, and the overall flow in the following sections.

A. Reward Computation

Assuming a system with p cores, c channels, and r ranks with p/c cores mapped to each channel and r/c ranks in each channel, we calculate the *reward* of activating a channel ch in the current interval as the ratio of its *value* to its *weight* in the interval when it was *last* activated, as shown in Equation 2. The channel value is measured as the sum of IPC of the cores (mapped to this channel) in its last active interval. The channel weight is the sum of the total power of all the ranks belonging to the channel in its last active interval. The power components of a memory rank j include its dynamic power ($P_{dyn}(j)$), leakage power ($P_{leak}(j)$), and refresh power ($P_{ref}(j)$), and are computed online as follows: (1) $P_{dyn}(j)$ is computed as the product of memory access counts of j and energy per read/write access, divided by the time interval, (2) $P_{leak}(j)$ is temperature dependent, and obtained from a lookup table (Section V-G), and (3) $P_{ref}(j)$ is taken to be a constant.

$$Reward_{ch} = \frac{\sum\limits_{i \in cores(ch)} IPC(i)}{\sum\limits_{j \in ranks(ch)} (P_{dyn}(j) + P_{leak}(j) + P_{ref}(j))} \tag{2}$$

B. Incorporating Adjacency-awareness

We identify three regions representing the *thermal state* of 3D DRAM during the workload execution, based on the maximum current memory temperature (T_{max}):

1) the *cool* region with $T_{max} < T_{cool}$, where T_{cool} denotes the temperature threshold for the *cool* region,
2) the *hot* region with $T_{cool} \leq T_{max} < T_{hot}$, where T_{hot} denotes the the temperature threshold for the *hot* region, and
3) the *critical* region with $T_{hot} \leq T_{max} \leq T_{crit}$

In the *cool* region, our *3D-TemPo* policy performs power allocation based on *memory activity rate* of channels in their last active interval. As thermal emergency does not occur in this region, prioritizing higher access frequency channels is most beneficial.

In the *hot region*, the channels are susceptible to thermal stalls, so *3D-TemPo* uses the *channel reward* (Eq. 2) considering both IPC of the cores and the DRAM's thermal condition. The channel rewards are sorted in the non-increasing order, and the channels are activated in that order.

To prevent vertically aligned thermal hotspots and improve the cooling efficiency we leverage adjacency-awareness in *3D-TemPo* for the *critical* region of 3D DRAM. The adjacency-awareness additionally considers the observed thermal correlation between vertically adjacent channels. While allocating power to the channels in the order of high channel reward, the policy skips the channels that are vertically adjacent to any of the already activated channels. This prevents the trapping of heat between vertically adjacent channels, minimizing DTM penalty. We skip a high reward channel only when its vertically adjacent neighbours n are in the *critical region* ($T_{hot} \leq T_n \leq T_{crit}$), where T_n denotes the neighbouring channel temperatures.

Figure 6 illustrates the adjacency-aware 3D-TemPo policy, with channel rewards listed at the bottom of 3D DRAM stack for the time instances t and t' and the memory being in the *critical* region. At time t, top reward Channels 3 and 0 are activated, and Channel 5 is skipped, as its vertically adjacent neighbor Channel 3, having temperature T_{ch3}, is in the *critical*

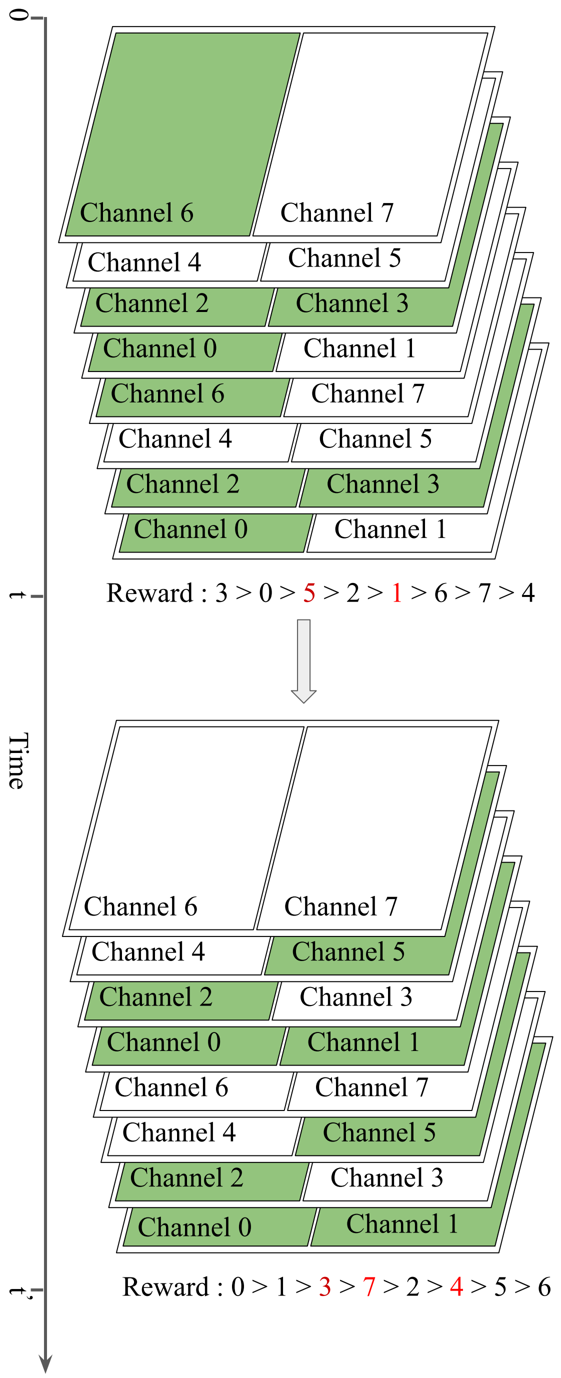


Fig. 6: Working of *Adjacency-aware 3D-TemPo* policy in the *critical* region. *Shading* represents *activated* channels.

region ($T_{ch3} > T_{hot}$). However, Channel 2, vertically adjacent to activated Channel 0, having temperature T_{ch0}, is not skipped as $T_{ch0} < T_{hot}$. Similarly, at time t', high reward channels 3, 7 and 4 are skipped due to Channels 1 and 2 being in the critical region ($T_{ch1} > T_{hot}$ and $T_{ch2} > T_{hot}$). To prevent the starvation of low reward channels, 3D-TemPo activates them once at every coarser time interval, ignoring the channel rewards.

Unlike *Round Robin*, *3D-TemPo* prioritizes *compute-heavy* cores, ensuring maximum system progress with minimum memory power dissipation. Unlike *MFU*, it selects the channels based on their *rewards* and not merely on the instantaneous memory throughput. We incorporate a channel's physical location in the 3D stack by considering the temperature-dependent leakage power. Top channels undergo a slow temperature rise, consuming less leakage power than the bottom channels even when subjected to similar memory access rates. *3D-TemPo* prioritizes top channels over bottom ones when similar *progress* is expected in the corresponding cores, reducing thermal stalls.

C. DRAM Low-power Based DTM

As discussed earlier, modern DRAMs allow rank-level control on the power states to help manage power dissipation. We employ the DRAM power states to also keep the memory channel temperatures under the thermal limit. Our policy performs the DTM at the channel level, whereby, a channel is sent to low-power *standby* state (causing a thermal stall) if any of its ranks exceeds T_{crit}. Similarly, a channel is reverted to *read/write* state upon cooling down of all its ranks. During thermal emergencies, the power budgeting policy does not consider the channel for power allocation regardless of the associated reward or priority. The DTM policy sends the temperature of memory channels to the budgeting policy at the start of each interval.

D. The Overall Flow

The *3D-TemPo* policy (Algorithm 1) uses low-power states based DTM for ensuring safe thermal limits and performs reward-based and adjacency-aware power allocation to memory channels, leveraging the thermal gradient in 3D DRAM. The workload execution is divided into time intervals (epochs), and at the beginning of every epoch, the maximum channel temperature T_{max} is obtained (Line 3). If one or more channels have exceeded T_{crit} (temperature constraint), the DTM policy is invoked and appropriate rank states (*RankState*) are computed (Lines 4-6). Potentially available channels that are not yet set to *standby* state, are activated in order of high memory activity in the *cool* region (Lines 7-8), and the remaining power budget is updated (Lines 9-11) . In the *hot* or *critical* region, the channel reward is computed and sorted in non-increasing order (Lines 13-15). Based on the available power budget P_b, the top reward channels are selected, checked for *critical* vertically adjacent neighbours (temperature $\geq T_{hot}$), activated upon *non-critical* vertical neighbours, and skipped otherwise, updating the remaining power budget (Lines 16-22). Upon cooling down of all channel ranks below T_{rec} (recovery temperature), indicating the heated channel has cooled down, the channel is reset to *read/write* state from *standby* (Line 23). Finally, the rank states and the activated channels are returned for the current epoch (Line 24) and sent to the memory controller for appropriate action.

V. Experimental Evaluation

A. Simulation Environment

We use an integrated performance-thermal simulator, CoMeT [53], consisting of Sniper 7.2 [54] multicore simulator

Algorithm 1: Adjacency-aware 3D-TemPo Policy

```
Input: $P_b$: Memory power budget
Input: $T[0:(c-1)]$: Memory channel temperatures
Input: $T_{crit}$: DTM invocation temperature
Input: $T_{rec}$: Recovery temperature
Input: $T_{cool}$: Threshold temperature for cool region
Input: $T_{hot}$: Threshold temperature for vertical
       neighbours
Output: $RankState[0:(r-1)]$: Power state of ranks,
        $Activated[0:(c-1)]$: Channel status
// Initialization
$Activated \leftarrow 0$
$P_{ActiveChannels} \leftarrow 0$
$T_{max} \leftarrow$ Get_Max_Channel_Temperature ($T$)
if $T_{max} > T_{crit}$ then // Heated. Apply DTM.
    Invoke_DTM_Policy ()
    // Place hot channels in standby.
    $RankState \leftarrow$ Rank_Power_State ($T$, $T_{crit}$)
else if $T_{max} < T_{cool}$ then // Cool region.
    $ActiveChannels \leftarrow$ Activate_Freq_Channels ($P_b$,
      $Activated$)
    for each channel ch in $ActiveChannels$ do
        $P_{ActiveChannels}$ += $P_{ch}$
    // Subtract power of channels in
       $ActiveChannels$ from Power Budget.
    $P_b$ -= $P_{ActiveChannels}$
else // Hot or critical region.
    for each channel ch in {0, 1, ... , (c-1)} do
        // Compute reward using Eq. 2
        $Reward[ch] \leftarrow$ Get_Channel_Reward ($ch$)
    $TopRewardChannels \leftarrow$ Sort_Rewards
      ($Reward$)
    for each channel ch in $TopRewardChannels$ do
        // Check if vertically adjacent
           neighbours in critical
           region.
        $Adj \leftarrow$ Is_Neighbour_Critical ($ch$, $T_{hot}$)
        // Non-critical vertical
           neighbours.
        if $Adj == False$ then
            $P_{ch} \leftarrow$ Compute_Req_Power ($ch$)
            if $P_b \geq P_{ch}$ then
                // Activate the channel.
                Activate_Channel ($ch$, $Activated$)
                // Update Power Budget.
                $P_b$ -= $P_{ch}$
// Place recovered channels to
   read/write state.
$RankState \leftarrow$ Set_Active_On_Recovery ($T$,$T_{rec}$)
return $RankState$, $Activated$
```

TABLE I: Simulation Workloads

Suite	Selected Benchmarks	Name	Type
SPEC 2017	lbm(×32)	WK-1	memory
	mcf(×32)	WK-2	memory
	nab(×16), x264(×8), exchange(×8)	WK-3	compute
(single-	exchange(×8), nab(×8), lbm(×16)	WK-4	mixed
threaded)	lbm(×24), gcc(×8)	WK-5	memory
	mcf(×8), lbm(×24)	WK-6	memory
	nab(×8), mcf(×8), gcc(×8), lbm(×8)	WK-7	mixed
	x264(×16), exchange(×16)	WK-8	compute
	lbm(×16), mcf(×16)	WK-9	memory
	gcc(×8), x264(×8), lbm(×8), exchange(×8)	WK-10	mixed
PARSEC 2.1	blackscholes(×32)	WK-11	compute
	bodytrack(×32)	WK-12	mixed
	fluidanimate(×32)	WK-13	mixed
(multi-	streamcluster(×32)	WK-14	memory
threaded)	swaptions(×32)	WK-15	compute

and Hotspot 6.0 [55], for running the experimental workloads, collecting memory access counts, and performing power allocation to channels every 1 ms (epoch time, *E*). We obtain the refresh and dynamic power dissipation using energy-per-access values (24.45 nJ per 64-byte access) from CACTI-3DD [56] and feed the power values to HotSpot thermal simulator with default configuration parameters [16]. Computed temperatures are sent back to Sniper, where the dynamic power budgeting and thermal management decisions are implemented.

We model a multi-core processor (32 cores, 3.6GHz, 22 nm, out-of-order, caches: 32KB private L1, 256 KB private L2, 32 MB shared L3) with an off-chip 3D DRAM (8GB size, 8 channels, 2 pseudo-channels per channel, 16 ranks/pseudo-channel, 1 bank/rank, 29 ns latency, and 44 GBps per channel bandwidth), and running workloads comprising 32 applications/threads, one on each core. The *standby* state consumes 17% static power and the transition overhead to *active* state is ∼6μs [48], negligible compared to the epoch time. We empirically determine the temperature thresholds by using different values and observing the performance gains: T_{cool}=74°C, T_{hot}=78°C, T_{rec}=77°C, and T_{crit}=80°C (similar to [16]).

We use a diverse set of workloads from *SPEC CPU2017* and *PARSEC 2.1* benchmark suites to validate the efficacy of our proposal. Table I presents the workload details and their characteristics. We attempted to cover a wide range of benchmark mixes with varying memory access characteristics. We simulate the compiled source code for PARSEC 2.1 workloads with input size *simlarge* and pre-generated traces (Pinballs) for 100M instructions for SPEC CPU2017.

B. Performance Improvement

We evaluate the performance of our *3D-TemPo* policy for two different power budgets, 64W and 96W, accounting for 50% and 75% of peak power consumption for our modelled architecture and simulated workloads, and a thermal limit of 80°C. Figures 7 and 8 show the execution time of different policies, normalized to *NoCons*, where *NoCons* represents a (hypothetical) case of running the workloads under no power and thermal constraints. We compare the performance of *3D-TemPo* against five baseline policies: (1) *MigrationBased*, an extended version of *FastCool* [16], (2) *RoundRobin*, (3) *Alternation*, a memory power budgeting variant of [29], (4)

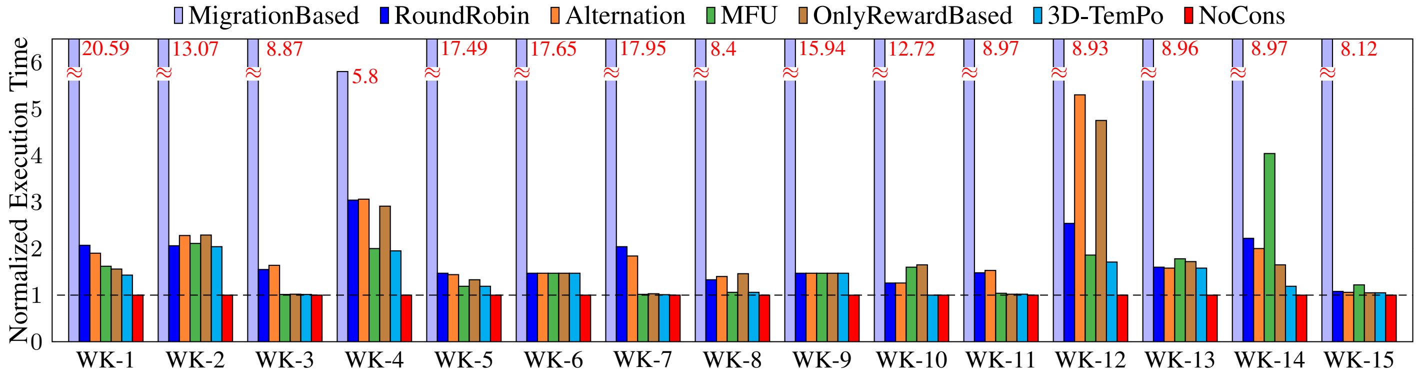


Fig. 7: Execution time of workloads with different power budgeting policies for P_b=64W and T_{crit}=80°C, normalized to no power and thermal constraints (NoCons).

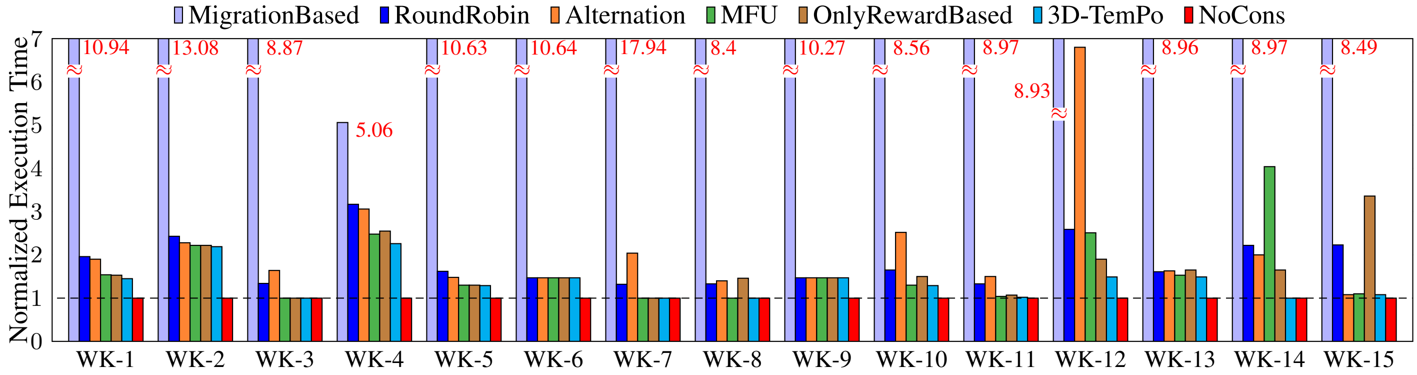


Fig. 8: Execution time of workloads with different power budgeting policies for P_b=96W and T_{crit}=80°C, normalized to no power and thermal constraints (NoCons).

Most Frequently Used (MFU), and (5) *OnlyRewardBased* component of *3D-Tempo*.

We report the execution time improvement of *3D-TemPo* over the baseline policies in Figures 7 and 8. *MigrationBased* performs poorly under limited power budget due to the associated data migration costs. We consistently observe better or similar performance with *3D-TemPo* for all workloads and all baselines, achieving speedups of 1x - 2.01x over *RoundRobin*, 1x - 3.09x over *Alternation*, 1x - 3.39x over *MFU*, 1x - 2.77x over *OnlyRewardBased* component, and 2.9x - 17.77x over *MigrationBased* for a 64W power budget. Similarly, we observe speedups of 1x - 2.24x over *RoundRobin*, 1x - 4.55x over *Alternation*, 1x - 4.07x over *MFU*, 1x - 3.1x over *OnlyRewardBased*, and 2.23x - 17.94x over *MigrationBased* for a 96W budget that allows more channel activations. *3D-TemPo* dynamically adapts to the workload phases by computing rewards and leveraging adjacency-awareness, minimizing the overall thermal impact.

C. Memory Energy Improvement

Figures 9 and 10 show the memory energy consumption (normalized to *NoCons*) of different policies for 64W and 96W power budgets respectively. *3D-TemPo* results in similar or reduced energy consumption compared to all baseline policies. Compared to *MigrationBased*, *3D-TemPo* does not consume migration energy, showing an energy dissipation ratio of 4.82x - 16.29x for 64W and 3.63x - 16.26x for 96W power budget. *3D-TemPo* activates channels based on their rewards and also eliminates vertically aligned thermal hotspots. This helps to reduce the attained channel temperatures and therefore the leakage power dissipation, leading to lower memory energy consumption. *MFU* prefers to activate channels with high memory activity with high dynamic power dissipation, causing increased memory energy consumption for memory-intensive and mixed workloads than *3D-TemPo*. *RoundRobin* and *Alternation*, which pick channels through rotation, always activate more number of channels simultaneously compared to *3D-TemPo* and consume similar or higher energy for all workloads. Compared to *OnlyRewardBased* policy, *3D-TemPo* reaches lower temperatures, and therefore, lower leakage power and memory energy consumption for memory-intensive and mixed workloads. Compared to other baseline policies, *3D-TemPo* provides a memory energy ratio of 1x - 1.73x over *RoundRobin*, 1x - 3.06x over *Alternation*, 1x - 3.08x over *MFU*, and 1x - 2.73x over *OnlyRewardBased* policy for a 64W budget. Similarly, for a 96W budget, *3D-TemPo* leads to memory energy ratio of 1.01x - 2.26x over *RoundRobin*, 1.02x - 4.33x over *Alternation*, 1x - 5.38x over *MFU*, and 1x - 2.45x over *OnlyRewardBased*.

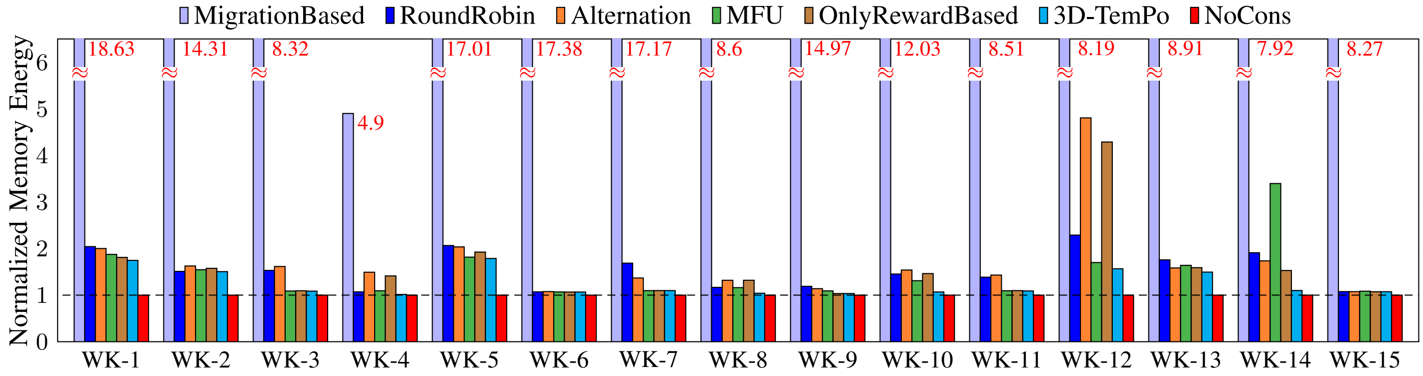


Fig. 9: Memory energy consumption of workloads with different power budgeting policies for P_b=64W and T_{crit}=80°C, normalized to no power and thermal constraints (NoCons).

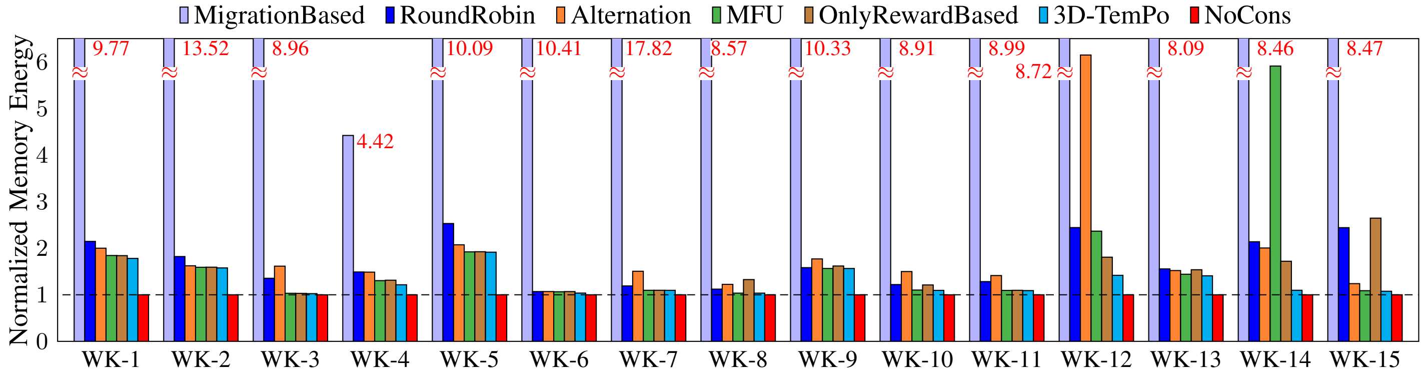


Fig. 10: Memory energy consumption of workloads with different power budgeting policies for P_b=96W and T_{crit}=80°C, normalized to no power and thermal constraints (NoCons).

D. Analysis of Policy Behavior

Observation 1: The order of channel activation/ deactivation is important. Workloads comprising *multi-threaded* compute-heavy applications (e.g., WK-11 and WK-15), with each thread running on a separate core, exhibit sub-optimal performance when all threads are treated uniformly. The channel temperatures remain far below T_{crit}, not requiring DTM. Policies such as *3D-TemPo* and *MFU*, prioritising high yielding threads, result in the best performance. The *single-threaded* compute-heavy workloads (eg., WK-3 and WK-8) also benefit equally from *3D-TemPo* and *MFU*.

Observation 2: The order of activation of vertically adjacent channels is important. Workloads comprising *memory-intensive* applications (single or multi-threaded) with high memory access rates suffer from severe DTM induced penalty (e.g., WK-1, WK-2, WK-5, and WK-14). Selecting a channel amongst the potentially heated vertically adjacent channels in the order of *high reward* ensures maximum progress and eliminates vertical thermal hotspots. Thus, the adjacency-aware *3D-TemPo* performs best for such workloads.

Observation 3: The scheduling of applications on processing cores is important. In workloads comprising *mixed* applications (e.g., WK-4, WK-7, WK-10, WK-12, and WK-13) or different *memory-intensive* applications with diverse memory-access rates (WK-6 and WK-9), core scheduling decides the application-to-channel mapping. A schedule that maps the compute tasks on bottom channels and memory-intensive ones on top channels exhibits lower temperatures and vice-versa. Interleaving compute and memory tasks on channels results in moderate heating. We observe that different schedules benefit differently from the same policy.

E. Transient Temperature Behavior

Figure 11 shows the transient temperature behavior of a *mixed* workload (WK-4) for different policies. The workload undergoes heating/cooling cycles due to intense memory activity of *lbm* application. We use an interleaved schedule for the workload, such that the compute and memory applications are interleaved across the memory channels. The duration between successive stalls varies with the order of channel activation in the policies. The *NoCons* case undergoes temperatures as high as 140°C. The power budgeting policies are able to prevent thermal violations; however, the number of thermal stalls and cooldown times are widely varying, as shown in Table II. Compared to *OnlyRewardBased* which prioritizes channel rewards even in the DRAM's *critical* region, *3D-TemPo* significantly reduces thermal stalls by avoiding the vertical thermal hotspots in the *critical* region. For the same reason, *OnlyRewardBased* only marginally outperforms *Alternation*, and *RoundRobin* policies. *MFU* and *3D-TemPo* have

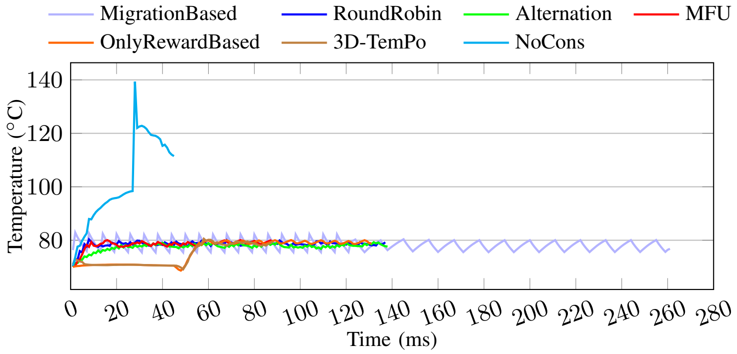


Fig. 11: Transient temperature for *WK-4* with different power budgeting policies for P_b=64W.

Power Budgeting	#Thermal Stalls	Avg. Cooldown Time
MigrationBased	18	16ms (includes migration delay)
RoundRobin	16	8ms
Alternation	28	5ms
MostRecentlyUsed	6	5ms
OnlyRewardBased	19	6ms
3D-TemPo	5	5ms
NoCons	NA	NA

TABLE II: Number of Thermal Stalls and Average Cooldown Time for WK-4

similar performance due to the interleaved schedule, which reduces the workload's thermal impact and consequently the penalty of enabling high frequency channels. *MigrationBased* undergoes frequent data migration to 2D memory due to power budgeting and DTM, and incurs maximum overheads in terms of *CoolDown* time.

F. Discussion on Working of Policies

Figure 12 shows the comparison of the working of different simplistic power budgeting policies (*RoundRobin*, *Alternation*, and *MFU*) and the reward based policies (*OnlyRewardBased* and *3D-TemPo*). We discuss the power budgeting mechanism of different policies during the execution of a memory-intensive workload WK-5 (lbm(×24), gcc(×8)) for a power budget P_b=64W (50% of the peak power consumption) and a thermal limit $T_{critical}$=80°C. We compare the policies for workload WK-5 on the basis of three attributes over time:

1) the number of simultaneously activated channels,
2) the utilization of the power budget, and
3) the peak memory temperature

On one hand, *RoundRobin* and *Alternation* policies activate 8 channels simultaneously during most of the execution merely by rotating amongst the channels. Due to this, these policies rarely utilize the full power budget and operate much below the permissible thermal limit, resulting in sub-optimal performance. On the other hand, *MFU* chooses to activate channels based on maximum memory activity (and therefore maximum dynamic power dissipation), utilizing the power budget very well and causing minimum wastage. For a memory-intensive workload such as WK-5, *MFU* performs reasonably well, however, the peak memory temperature remains close to the thermal limit and the memory undergoes thermal stalls. *OnlyRewardBased* policy prioritizes channels based on channel rewards that also affect power budget utilization. Activating high reward channels without avoiding the accumulation of vertically aligned hotspots results in high peak memory temperatures and consequently thermal stalls and lesser number of channels available for activation. *3D-TemPo* intelligently selects the channels for activation, utilizes the power budget well, and is able to deliver the highest system performance by activating 8 *best* channels for most part of workload execution. By choosing the channels that guarantee maximum progress and also contribute minimum thermal footprint, *3D-TemPo* outruns all other policies. Due to not activating vertically aligned hot neighbors, *3D-TemPo* prevents trapping of heat, resulting in reduced peak memory temperature.

G. Implementation Details

We implement *3D-TemPo* policy as a software mechanism that periodically sends the rank power states to the memory controller. To measure channel temperatures, we assume the placement of two thermal sensors at each DRAM die [1]. The temperature-dependent leakage power of a channel is looked up from a table that stores P_{leak} at 10°C-wide temperature ranges obtained using CACTI-3DD [56]. Our workloads exhibit temperatures in $[60°C - 80°C]$ requiring only two table entries, and hence, negligible storage and lookup time. Computing P_{dyn}, which uses the DRAM access count per channel, requires one multiply operation. We estimate the time overhead of *3D-TemPo* by running it on a simulated core, observing a maximum delay of 12μs, which is negligible compared to the epoch time.

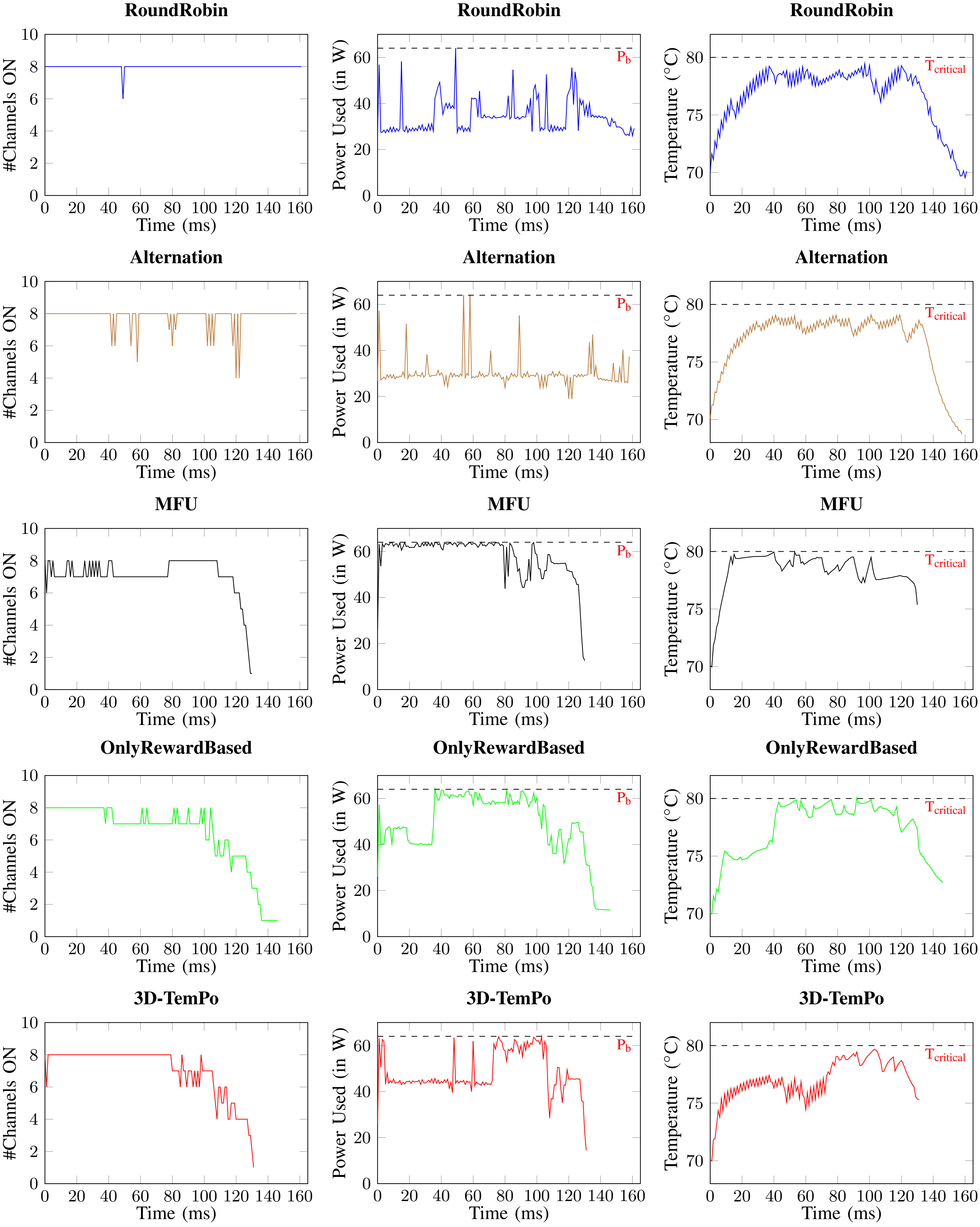
VI. Conclusion

3D DRAMs are often limited by their power budgets and thermal constraints, resulting in under-utilization of high memory bandwidth. Simplistic power budgeting policies, unaware of physical location of channels and favouring a single metric, do not result in the best performance. We present a heuristic for determining a channel's reward that efficiently captures the system's progress on activating the channel. Further, we leverage the adjacency-awareness to minimize associated overheads of power budgeting and DTM. Our results show speedups of 1x to 17.94x over the baseline policies. In the future, we plan to investigate prediction-based and application-aware budgeting policies for 3D DRAMs.

References

[1] JEDEC, "JEDEC Standard High Bandwidth Memory DRAM (HBM3), JESD238," 2022. [Online]. Available: https://www.jedec.org/standards-documents/docs/jesd238

[2] Micron, "Micron Hybrid Memory Cube – HMC Gen," 2018. [Online]. Available: https://www.micron.com/-/media/client/global/documents/products/data-sheet/hmc/gen2/hmc_gen2.pdf



(a)

(b)

(c)

Fig. 12: Comparison of (a) number of channels in ON state over time, (b) power budget utilization over time, and (c) peak memory temperature over time in different power budgeting policies.

[3] J. Byrne, "Powerful Hardware and a Strong Software Ecosystem Help Layerscape Excel at AI," 2018. [Online]. Available: https://www.nxp.com/

[4] Intel, "Intel® Xeon® Max Series CPUs." 2022. [Online]. Available: {https://www.intel.com/content/www/us/en/products/docs/processors/max-series/overview.html}

[5] J. Liu, B. Jaiyen, Y. Kim, C. Wilkerson, and O. Mutlu, "An experimental study of data retention behavior in modern dram devices: Implications for retention time profiling mechanisms," vol. 41, no. 3, p. 60–71, jun 2013. [Online]. Available: https://doi.org/10.1145/2508148.2485928

[6] T. Hamamoto, S. Sugiura, and S. Sawada, "On the retention time distribution of dynamic random access memory (dram)," *IEEE Transactions on Electron Devices*, vol. 45, no. 6, pp. 1300–1309, 1998.

[7] W.-H. Lo, K.-z. Liang, and T. Hwang, "Thermal-aware dynamic page allocation policy by future access patterns for hybrid memory cube (hmc)," in *Design, Automation & Test in Europe Conf. & Exhibition*. IEEE, 2016, pp. 1084–1089.

[8] S. Kim, W. Kwak, C. Kim, D. Baek, and J. Huh, "Charge-aware dram refresh reduction with value transformation," in *2020 IEEE International Symposium on High Performance Computer Architecture (HPCA)*, 2020, pp. 663–676.

[9] Y.-C. Kwon, S. H. Lee, J. Lee, S.-H. Kwon, J. M. Ryu, J.-P. Son, O. Seongil, H.-S. Yu, H. Lee, S. Y. Kim, Y. Cho, J. G. Kim, J. Choi, H.-S. Shin, J. Kim, B. Phuah, H. Kim, M. J. Song, A. Choi, D. Kim, S. Kim, E.-B. Kim, D. Wang, S. Kang, Y. Ro, S. Seo, J. Song, J. Youn, K. Sohn, and N. S. Kim, "25.4 a 20nm 6gb function-in-memory dram, based on hbm2 with a 1.2tflops programmable computing unit using bank-level parallelism, for machine learning applications," in *2021 IEEE International Solid- State Circuits Conference (ISSCC)*, vol. 64, 2021, pp. 350–352.

[10] J. Ahn, S. Yoo, and K. Choi, "Low-power hybrid memory cubes with link power management and two-level prefetching," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 24, no. 2, pp. 453–464, 2016.

[11] H. Esmaeilzadeh, E. Blem, R. S. Amant, K. Sankaralingam, and D. Burger, "Dark silicon and the end of multicore scaling," in *2011 38th Annual International Symposium on Computer Architecture (ISCA)*, 2011, pp. 365–376.

[12] M. B. Taylor, "Is dark silicon useful? harnessing the four horsemen of the coming dark silicon apocalypse," in *DAC Design Automation Conference 2012*, 2012, pp. 1131–1136.

[13] N. Hardavellas, M. Ferdman, B. Falsafi, and A. Ailamaki, "Toward dark silicon in servers," *IEEE Micro*, vol. 31, no. 4, pp. 6–15, 2011.

[14] M. B. Taylor, "A landscape of the new dark silicon design regime," *IEEE Micro*, vol. 33, no. 5, pp. 8–19, 2013.

[15] N. Sayed, S. M. Nair, R. Bishnoi, and M. B. Tahoori, "Process variation and temperature aware adaptive scrubbing for retention failures in stt-mram," in *2018 23rd Asia and South Pacific Design Automation Conference (ASP-DAC)*, 2018, pp. 203–208.

[16] L. Siddhu, R. Kedia, and P. R. Panda, "Leakage-aware dynamic thermal management of 3d memories," *ACM Trans. on Design Automation of Electronic Syst. (TODAES)*, vol. 26, no. 2, pp. 1–31, 2020.

[17] X. Zhou, J. Yang, Y. Xu, Y. Zhang, and J. Zhao, "Thermal-aware task scheduling for 3d multicore processors," *IEEE Transactions on Parallel and Distributed Systems*, vol. 21, no. 1, pp. 60–71, 2010.

[18] A.-C. Hsieh and T. Hwang, "Thermal-aware memory mapping in 3d designs," in *2009 Design, Automation Test in Europe Conference Exhibition*, 2009, pp. 1361–1366.

[19] S. P. Muralidhara, L. Subramanian, O. Mutlu, M. Kandemir, and T. Moscibroda, "Reducing memory interference in multicore systems via application-aware memory channel partitioning," in *IEEE/ACM Int'l. Symp. on Microarchitecture*. IEEE, 2011, pp. 374–385.

[20] T. Zhang, M. Poremba, C. Xu, G. Sun, and Y. Xie, "Cream: A concurrent-refresh-aware dram memory architecture," in *2014 IEEE 20th International Symposium on High Performance Computer Architecture (HPCA)*, 2014, pp. 368–379.

[21] K. Sudan, K. Rajamani, W. Huang, and J. B. Carter, "Tiered memory: An iso-power memory architecture to address the memory power wall," *IEEE Transactions on Computers*, vol. 61, no. 12, pp. 1697–1710, 2012.

[22] T. S. Muthukaruppan, M. Pricopi, V. Venkataramani, T. Mitra, and S. Vishin, "Hierarchical power management for asymmetric multi-core in dark silicon era," in *2013 50th ACM/EDAC/IEEE Design Automation Conference (DAC)*, 2013, pp. 1–9.

[23] H. Khdr, S. Pagani, M. Shafique, and J. Henkel, "Thermal constrained resource management for mixed ilp-tlp workloads in dark silicon chips," in *2015 52nd ACM/EDAC/IEEE Design Automation Conference (DAC)*, 2015, pp. 1–6.

[24] H. Wang, M. Zhang, S. X.-D. Tan, C. Zhang, Y. Yuan, K. Huang, and Z. Zhang, "New power budgeting and thermal management scheme for multi-core systems in dark silicon," *2016 29th IEEE International System-on-Chip Conference (SOCC)*, pp. 344–349, 2016.

[25] G. Kornaros and D. Pnevmatikatos, "Dynamic power and thermal management of noc-based heterogeneous mpsocs," vol. 7, no. 1, feb 2014. [Online]. Available: https://doi.org/10.1145/2567658

[26] O. Sahin and A. K. Coskun, "On the impacts of greedy thermal management in mobile devices," *IEEE Embedded Systems Letters*, vol. 7, no. 2, pp. 55–58, 2015.

[27] G. Bhat, G. Singla, A. K. Unver, and U. Y. Ogras, "Algorithmic optimization of thermal and power management for heterogeneous mobile platforms," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 26, no. 3, pp. 544–557, 2018.

[28] G. Singla, G. Kaur, A. K. Unver, and U. Y. Ogras, "Predictive dynamic thermal and power management for heterogeneous mobile platforms," in *2015 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 2015, pp. 960–965.

[29] H. Wang, W. He, Q. Yang, X. Peng, and H. Tang, "Dbp: Distributed power budgeting for many-core systems in dark silicon," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 41, no. 12, pp. 5727–5731, 2022.

[30] A. Singh, C. Leech, K. R. Basireddy, B. Al-Hashimi, and G. Merrett, "Learning-based run-time power and energy management of multi/many-core systems: Current and future trends," *Journal of Low Power Electronics*, vol. 13, 06 2017.

[31] T. Ebi, D. Kramer, W. Karl, and J. Henkel, "Economic learning for thermal-aware power budgeting in many-core architectures," in *2011 Proceedings of the Ninth IEEE/ACM/IFIP International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS)*, 2011, pp. 189–196.

[32] T. Ebi, M. A. Al Faruque, and J. Henkel, "Tape: Thermal-aware agent-based power economy for multi/many-core architectures," in *Proceedings of the 2009 International Conference on Computer-Aided Design*, ser. ICCAD '09. New York, NY, USA: Association for Computing Machinery, 2009, p. 302–309. [Online]. Available: https://doi.org/10.1145/1687399.1687457

[33] 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 Transactions on Computers*, vol. 68, no. 4, pp. 526–541, 2019.

[34] S. Niknam, A. Pathania, and A. D. Pimentel, "T-tsp: Transient-temperature based safe power budgeting in multi-/many-core processors," in *2021 IEEE 39th International Conference on Computer Design (ICCD)*, 2021, pp. 500–508.

[35] M. Rapp, M. Sagi, A. Pathania, A. Herkersdorf, and J. Henkel, "Power- and cache-aware task mapping with dynamic power budgeting for many-cores," *IEEE Transactions on Computers*, vol. 69, no. 1, pp. 1–13, 2020.

[36] X. Wang, B. Zhao, L. Wang, T. Mak, M. Yang, Y. Jiang, and M. Daneshtalab, "A pareto-optimal runtime power budgeting scheme for many-core systems," *Microprocessors and Microsystems*, vol. 46, pp. 136–148, 2016. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0141933116300126

[37] A. Prakash, H. Amrouch, M. Shafique, T. Mitra, and J. Henkel, "Improving mobile gaming performance through cooperative cpu-gpu thermal management," in *Design Automation Conf. (DAC)*. IEEE, 2016, pp. 1–6.

[38] A. Pathania, Q. Jiao, A. Prakash, and T. Mitra, "Integrated cpu-gpu power management for 3d mobile games," in *Design Automation Conf. (DAC)*. IEEE, 2014, pp. 1–6.

[39] U. Gupta, R. Ayoub, M. Kishinevsky, D. Kadjo, N. Soundararajan, U. Tursun, and U. Y. Ogras, "Dynamic power budgeting for mobile systems running graphics workloads," *IEEE Transactions on Multi-Scale Computing Systems*, vol. 4, no. 1, pp. 30–40, 2018.

[40] C.-H. Liao, C. H.-P. Wen, and K. Chakrabarty, "An online thermal-constrained task scheduler for 3d multi-core processors," in *2015 Design, Automation Test in Europe Conference Exhibition (DATE)*, 2015, pp. 351–356.

[41] F. Hameed, M. A. A. Faruque, and J. Henkel, "Dynamic thermal management in 3d multi-core architecture through run-time adaptation," in *2011 Design, Automation Test in Europe*, 2011, pp. 1–6.

[42] D. Lee, S. Das, J. R. Doppa, P. P. Pande, and K. Chakrabarty, "Performance and thermal tradeoffs for energy-efficient monolithic 3d network-on-chip," *ACM Trans. Des. Autom. Electron. Syst.*, vol. 23, no. 5, aug 2018. [Online]. Available: https://doi.org/10.1145/3223046

[43] A. K. Coskun, J. L. Ayala, D. Atienza, T. S. Rosing, and Y. Leblebici, "Dynamic thermal management in 3d multicore architectures," in *2009*

Design, Automation & Test in Europe Conference & Exhibition, 2009, pp. 1410–1415.

[44] C. Zhu, Z. Gu, L. Shang, R. P. Dick, and R. Joseph, "Three-dimensional chip-multiprocessor run-time thermal management," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 27, no. 8, pp. 1479–1492, 2008.

[45] 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 *Design Automation Conf.* IEEE, 2012, pp. 648–655.

[46] M. K. Jeong, D. H. Yoon, D. Sunwoo, M. Sullivan, I. Lee, and M. Erez, "Balancing dram locality and parallelism in shared memory cmp systems," in *IEEE International Symposium on High-Performance Comp Architecture*, 2012, pp. 1–12.

[47] S. Pandey and P. R. Panda, "Neuromap: Efficient task mapping of deep neural networks for dynamic thermal management in high-bandwidth memory," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 41, no. 11, pp. 3602–3613, 2022.

[48] Y. Lu, D. Wu, B. He, X. Tang, J. Xu, and M. Guo, "Rank-aware dynamic migrations and adaptive demotions for dram power management," *IEEE Transactions on Computers*, vol. 65, no. 1, pp. 187–202, 2016.

[49] I. Hur and C. Lin, "A comprehensive approach to dram power management," in *2008 IEEE 14th International Symposium on High Performance Computer Architecture*, 2008, pp. 305–316.

[50] B. Akin, F. Franchetti, and J. C. Hoe, "Data reorganization in memory using 3d-stacked dram," *SIGARCH Comput. Archit. News*, vol. 43, no. 3S, p. 131–143, jun 2015. [Online]. Available: https://doi.org/10.1145/2872887.2750397

[51] A.-C. Hsieh and T. Hwang, "Thermal-aware memory mapping in 3d designs," *ACM Trans. on Embedded Computing Syst. (TECS)*, vol. 13, no. 1, pp. 1–22, 2013.

[52] M. J. Khurshid and M. Lipasti, "Data compression for thermal mitigation in the hybrid memory cube," in *2013 IEEE 31st International Conference on Computer Design (ICCD)*, 2013, pp. 185–192.

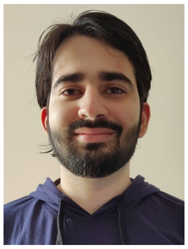
[53] L. Siddhu, R. Kedia, S. Pandey, M. Rapp, A. Pathania, J. Henkel, and P. R. Panda, "CoMeT: An integrated interval thermal simulation toolchain for 2D, 2.5D, and 3D processor-memory systems," *ACM Trans. Archit. Code Optim.*, 2022.

[54] T. E. Carlson, W. Heirman, S. Eyerman, I. Hur, and L. Eeckhout, "An evaluation of high-level mechanistic core models," *ACM Trans. Archit. Code Optim.*, vol. 11, no. 3, aug 2014. [Online]. Available: https://doi.org/10.1145/2629677

[55] M. R. S. R. Zhang and K. Skadron, "Hotspot 6.0: Validation, acceleration and extension, technical report cs-2015-04, university of virginia," 2015.

[56] K. Chen, S. Li, N. Muralimanohar, J. H. Ahn, J. B. Brockman, and N. P. Jouppi, "Cacti-3dd: Architecture-level modeling for 3d die-stacked dram main memory," in *2012 Design, Automation Test in Europe Conference Exhibition (DATE)*, 2012, pp. 33–38.

Shailja Pandey is a research scholar in the Department of Computer Science and Engineering at the Indian Institute of Technology Delhi, India. She obtained her Masters degree in Computer Science & Engineering from the Indian Institute of Technology Kharagpur, India in 2016. Her research interests include performance optimizations and thermal management in novel memory technologies.



Sayam Sethi is a final year undergraduate student in the Department of Computer Science and Engineering at the Indian Institute of Technology Delhi, India. He is interested in researching problems in the field of Computer Architecture, Embedded Systems and Security.

Preeti Ranjan Panda received his B. Tech. in Computer Science and Engineering from the IIT Madras in 1990 and his M.S. and Ph.D. from the University of California at Irvine in 1995 and 1998, respectively. He is currently a Professor in the Department of Computer Science and Engineering at IIT Delhi. He has previously worked at Texas Instruments and Synopsys, Inc., and was a visiting scholar at Stanford University. He is the author of two books and a recipient of IBM Faculty Award, IESA Techno Mentor Award, and DST Young Scientist Award. He serves as the Editor-in-Chief of IEEE Embedded Systems Letters and has served on the editorial boards of several journals including IEEE TCAD, ACM TODAES, and Springer IJPP. He has served as the Technical Program chair of CODES+ISSS and CASES, on the steering committee of ASPDAC, and on the Technical Program Committees of several major conferences including DAC, ICCAD, and DATE.