IJIRR

International Journal of Information Research and Review
Vol. 04, Issue, 08, pp.4415-4423, August, 2017

RESEARCH ARTICLE

RUNTIME THERMAL MANAGEMENT BASED ON A NOVEL TASK MIGRATION TECHNIQUE IN 3D CHIP MULTIPROCESSORS

***Sulaiman Aljeddani and Farah Mohammadi**

Electrical and Computer Engineering Department, Ryerson University, 350 Victoria Street, M5B 2K3, Toronto, ON, Canada

ARTICLE INFO

Article History:

Received 22nd May, 2017
Received in revised form
27th June, 2017
Accepted 04th July, 2017
Published online 30th August, 2017

Keywords:

3D CMPs;
Task migration; Hotspots;
Runtime thermal management;
Cache layer.

ABSTRACT

The Chip Multiprocessors (CMPs) architecture moves from multi-core to many-core architecture to provide higher computing performance, and more reliable systems. Moreover, the CMPs trend also move from 2D CMPs to 3D CMPs architecture in order to obtain higher performance, more reliability, reduced cache access latency, and increased cache bandwidth when compared with 2D CMPs. Therefore, in this work we present a 3D many-core CMP architecture which executes heavy loaded tasks in order to improve the system performance. However, executing heavy loaded tasks demands increasing in system power consumption which results in increasing the on-chip thermal hotspots. The thermal hotspots in the 3D many-core CMPs cause performance degradation, reducing reliability, decreasing the chip life spam. Therefore, Runtime Thermal Management (RTM) in the 3D many-core CMPs has become crucial to control the thermal hotspots without any performance degradation. In this paper, a new runtime task migration technique is proposed to control hotspots in the 3D many-core CMPs. The proposed technique migrates the hottest tile with the optimal coldest tile in the core layer. The optimal coldest tileis selected by considering theDynamic Random Access Memory (DRAM) banks' access distribution level in the cache layer. The simulation results indicate up to 33□ (on average 13□) reduction in the cores' temperature of the target 3D many-core CMP. Moreover, the proposed technique efficiency is clarified in the simulation results that the maximum temperature of cores in the core and cache layers are both less than the maximum temperature limit, 80.

INTRODUCTION

Current trends indicate that the embedded systems industry is to gradually increase the number of transistors in a single chip as the size of transistors is to shrink (Hassanpour, 2013 and Henkel, 2015). This change gives designers the ability to design more complicated circuits such as chip multiprocessors (CMPs). Furthermore, the aforementioned feature makes it possible to increase the number of cores in CMPs which causes them to obtain higher performance and more reliable systems (Liu, 2015). Over time, embedded systems have moved away from two dimensional integrated circuits (2D ICs) to three dimensional integrated circuits (3D ICs). The 3D ICs, when compared with 2D ICs designs, reduce interconnection wire length which results in lower power consumption and shorter communication latency. Therefore, by combining the 3D ICs technology and CMPs such that they become 3D CMPs they will obtain higher computing performance and more efficient and reliable systems. Moreover, the architecture of CMPs has been extended to the 3D CMPs' architecture by using through silicon vias (TSVs) (Coskun, 2006 and Cui, 2016). In this work, the target architecture is a 3D many-core CMP which is shown in Figure 1. In this context, the architecture of the target 3D many-core CMPs has the potential to achieve a number of benefits such as increased system performance, greater reliability, reduced memory access latency, and a larger amount of memory bandwidth when compared to the conventional 2D many-core CMPs (Zhao, 2013 and Sun, 2009). However, the advantages introduced above will be achieved by loading complex tasks on to the system which will result in increasing the system's power consumption. The increased power consumption will create higher temperature spots, also known as thermal hotspots (Heo, 2003). These hotspots lead to performance degradation, a decrease in reliability, a rise in cooling costs, a shortened life span of the circuit, and the eventual failure of the system (Liu, 2015; Murali, 2008; Asad, 2015). Therefore, Runtime Thermal Management (RTM) becomes necessary to control the thermal hotspots and thereby improving the performance of the 3D many-core CMPs (Hassanpour, 2013). However, migrating the heavy loaded task to a low temperature tile in the core layer in order to control the temperature variances without considering hotspots in the

Corresponding author: **Sulaiman Aljeddani**
Electrical and Computer Engineering Department, Ryerson University, 350 Victoria Street, M5B 2K3, Toronto, ON, Canada

stacked cache layer has the potential to cause the emergence of new hotspots (Zhao, 2013; Murali, 2007; Murali, 2008; Hanumaiah, 2014 and March, 2013). Therefore, a new runtime task migration technique is being proposed to control temperature variances both in the core layer and the cache layer simultaneously in order to make the optimum task migration decisions more efficiently. In this paper, the proposed technique leads to balanced hotspots and temperature variations on the 3D many-core CMPs. In this context, it is crucial that the system must select an optimal coldest tile to be migrated with the hottest tile in the core layer rather than selecting the coldest tile. The optimal coldest tile refers to a cold tile in the core layer that is not located under a hotspot Dynamic Random Access Memory (DRAM) bank. To the best of our knowledge, this is the first attempt at proceeding task migration by considering two stacked layers simultaneously. When considering hotspots in both layers, the results of the proposed algorithm should lead to a significant reduction of hotspots in the 3D many-core CMPs. Finally, the rest of this paper is organized as follows. The paper gives a summary of related works in Section 2. The target system architecture is described in Section 3. In Section 4, the proposed algorithm is evaluated and its significance is discussed. Experimental evaluation is presented in Section 5. Lastly, Section 6 draws the main conclusions of the proposed runtime task migration technique.

The target 3D many-core CMP architecture contains the core layer and the cache layer.

There are multicore techniques that have been presented in (Hassanpour, 2013; Zhao, 2013; Liu, 2015;0020 Murali, 2007; Murali, 2008 and Hanumaiah, 2014), for runtime thermal management which consist of dynamic voltage and frequency scaling (DVFS) (Rotem, 2009), task migration (Hassanpour, 2013; Zhao, 2013; Liu, 2015; Murali, 2007 and Murali, 2008), (Heo, 2003), and clock gating first that were developed for single-core processors (Kadjo, 2013). For instance, the DVFS technique in (Hanumaiah, 2014) and (Kim, 2008), can control the temperature by dynamically adjusting the processor speed based on the workload. However, DVFS techniques sacrifice the performance in order to cool down the temperature. In (Hassanpour, 2013; Zhao, 2013), the proposed algorithms, in some cases, are unable to find a proper destination core due to the thermal constraints, which resulted in the authors using DVFS which has proved to be inefficient as far as performance is concerned. Moreover, in (Zhao, 2013) authors implemented many thermal-aware algorithms to migrate threads between the cores in order to reduce thermal variation in the 3D architecture. However, some techniques involve proceed static thread migration which in some cases can migrate a thread from a cold core to a hotspot. Also in (Zhao, 2013 and Liu, 2015), the authors proposed other techniques which always assign the new job to the closest core to balance the thermal hotspots across the 3D chip, however, they increase hotspots in the system rapidly. In (Li, 2014; Kang, 2010; Tajik, 2013; Meng, 2012; Sun, 2009 and Rasquinha, 2010), a number of new green memory architectures to combat temperature-related problems

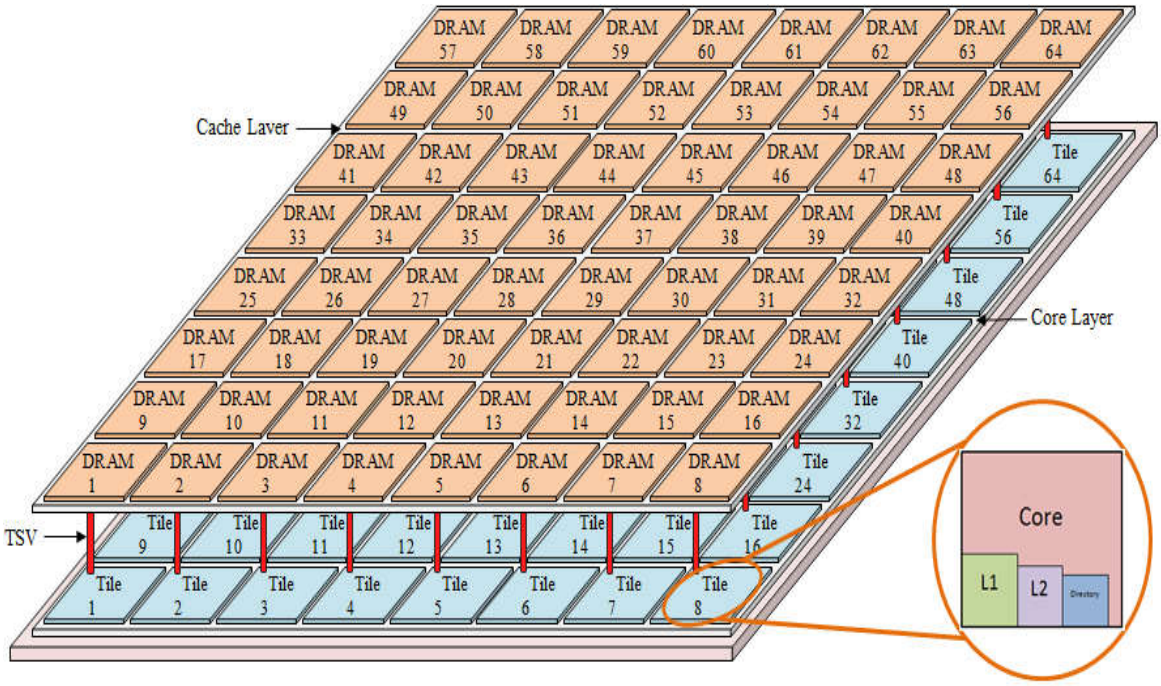

Figure 1. The target 3D many-core CMP architecture contains the core layer and the cache layer

Related Works

The goal of runtime thermal management is to keep CMPs working below the maximum threshold temperature while maintaining an e□cient task execution performance. In this case, off-line methods are not effective when hundreds of processors are integrated on advanced technologies. Thereby, on-chip runtime thermal management techniques are required to address the thermal issues of each core in each layer (Hassanpour, 2013; Zhao, 2013 and Liu, 2015).

in CMPs have been proposed. In CMPs research, a great deal of studies have focused on optimizing performance subject to temperature and power constraints (Murali, 2007; Murali, 2008; Zhan, 2014). The main disadvantage of these techniques is that they are based on design time and they cannot manage the thermal distribution within their systems with respect to changes in their workload. Voltage/frequency scaling, task scheduling (Chantem, 2011) and task allocation are other techniques for relieving hotspots in CMPs (Hanumaiah, 2014; Coskun, 2009). In this context, on-chip interconnection network

thermal management in CMPs is another topic that is addressed in (Hanumaiah, 2014 and Kang, 2011). Power management techniques with the goal of temperature reduction and a decrease in hot spots are other techniques that have been studied in recent years (Cheng, 2014 and Kadjo, 2013). Clock gating techniques are also used to manage temperature in CMPs (Kadjo, 2013). When a CMP system's cores run at the highest default frequency and voltage setting, a core will always reach the thermal threshold. At this time, the prevention of a hotspot core's clock is gated in order to prevent system failure and an occurring hotspot. In this work, a new runtime task migration technique which offers an effective solution to the thermal challenge in the 3D many-core CMPs is presented. The proposed technique considers hotspots effects in both core and cache layers simultaneously, in order to make the optimal task migration decisions for reducing hotspots and temperature variances (TempVar) on the 3D many-core CMPs.

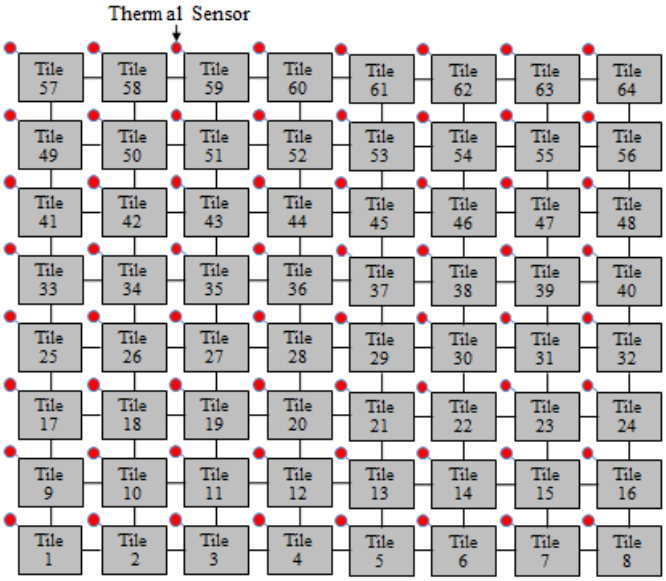

Figure 2. The thermal sensors' placement on each tile in the core layer

The Target System Architecture

In this paper, the target 3D many-core CMPs architecture includes two main parts as it shown in Figure 1. The first part is the down layer which is named the core layer. This layer includes 64 tiles. Each tile includes a core, a private L1 cache bank, and a shared cache L2 bank which is also illustrated in Figure 1. Moreover, each tile has a thermal sensor as shown in Figure 2, in order to measure its temperature. Furthermore, the second part of the target 3D many-core CMPs architecture is the upper layer which is named the cache layer. This layer includes 64 DRAM banks. Each DRAM bank also has a performance counter to calculate its accesses percentage level. Figure 3 shows each performance counters' connection with the DRAM banks. In this work, the proposed technique aims to obtain the most efficient task migration decision in order to minimize thermal impacts on the core layer and the cache layer simultaneously. Therefore, a centralized hardware named the Temperature Control Unit (TCU) is presented. TCU is the system decision maker, and it is responsible for the fulfillment of RTM in the 3D many-core CMPs. TCU is assumed to have been be placed near all of the tiles in the core layer. Figure 4 demonstrates the TCU and its connection with a tile. Moreover, the TCU has a table named TCU table. The TCU table stores the system information such as lists of each tile's temperature in the core layer and each tile's location. It also includes the accesses distribution to each DRAM bank in the cache layer and their locations. The TCU table is sorted in some steps that will be explained in the next section. An example of the sorted TCU table is illustrated in Figure 5. It is noticeable that statistical information is gathered from the entire processor and is sent to the TCU at the end of each time interval, fixed at 100ms in this work. The hardware overhead used in the target 3D many-core CMPs architecture is shown in Table I.

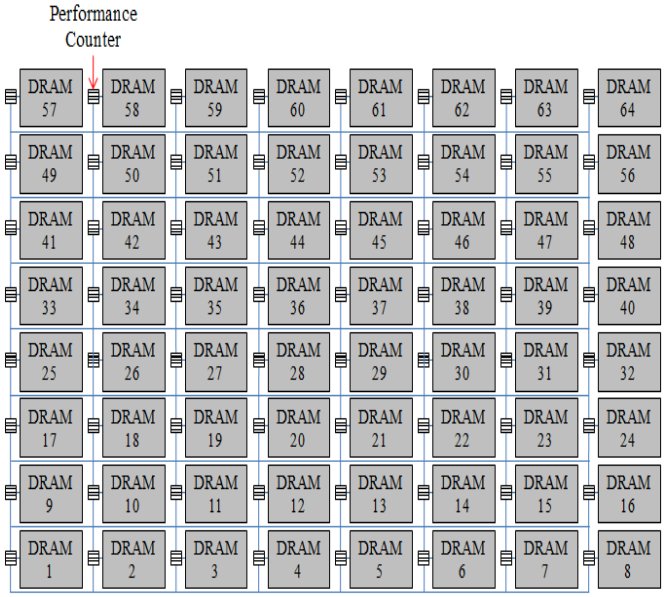

Figure 3. The performance counters' connection with DRAM banks in the cache layer

Table 1. The hardware overhead of the target 3d many-core cmps architecture

Hardware	Overhead
Performance counter	64*32bits
Thermal Sensors	64 sensors
TCU table	64* (8+8+4+4)bits

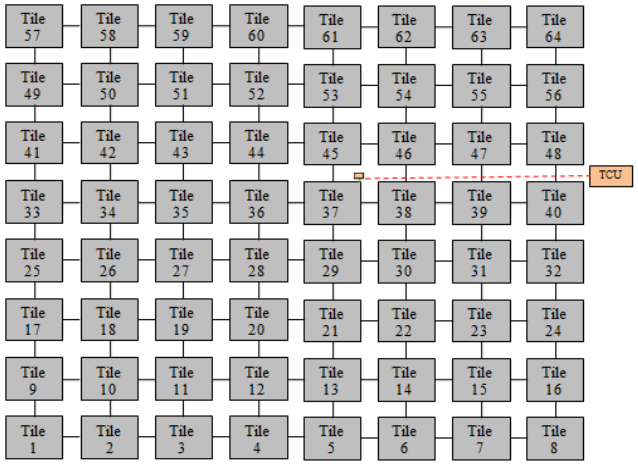

Figure 4. The TCU and its connection with a tile in the core layer

The Proposed Algorithm

The 3D many-core CMPs allows for the operation of heavy loaded tasks as a result of the shrinking transistor sizes and the

increasing number of cores on the chip. Even though the heavy loaded tasks provide a higher computing performance, they also increase power consumption. The increased power consumption leads to a rise in the temperature (hotspots) on the 3D many-core CMPs (Hassanpour, 2013; Zhao, 2013; Liu, 2015). To combat these temperature related problems, the proposed runtime task migration technique has become necessary in the future 3D many-core CMPs in order to control hotspots.

The Core Layer		The Cache layer	
Tile's Location	Tile's Temperature	DRAM's location	DRAM's accesses distribution
(3,0)	95°C	(3,0)	2
(1,0)	92°C	(1,0)	3
(0,0)	90°C	(0,0)	5
(1,1)	90°C	(1,1)	5
(0,1)	88°C	(0,1)	2
(2,2)	86°C	(2,2)	4
(3,3)	86°C	(3,3)	3
(0,2)	85°C	(0,2)	4
:	:	:	:
(2,0)	56°C	(2,0)	4
(1,2)	55°C	(1,2)	3
(2,1)	53°C	(2,1)	5
(3,1)	52°C	(3,1)	5
(0,3)	50°C	(0,3)	2
(3,2)	49°C	(3,2)	4
(2,3)	49°C	(2,3)	3
(1,3)	48°C	(1,3)	5

Figure 5. A sample of the sorted TCU table

In this paper, the important issue for starting the proposed migration technique is how to make the most efficient decision for finding the best tile to be migrated with the hottest tile in core layer. Therefore, the most important point is which tile amongst the cold tiles is the best to select. It is crucial that the system selects the optimal coldest tile instead of selecting the coldest tile. In fact, each tile has a related DRAM bank which is located on top of it. Thus, in order to find the optimal coldest tile, the TCU must consider the temperature of each tile in the core layer simultaneously with the percentage accesses level of its related DRAM bank in the cache layer. After that, TCU selects the optimal coldest tile. The selected optimal coldest tile must not have a most accessed related DRAM bank. If the TCU selects a cold tile where its related DRAM bank is a most accessed DRAM bank, then there is a possibility that the selected tile becomes a hotspot faster. In other words, selecting the optimal coldest tile in the core layer will prevent the selection of a cold tile that may be located under a most accessed DRAM bank and thus it will prevent the possible appearance of any new hotspots. Based on this trend, TCU should determine the hottest tile to be migrated with the optimal coldest tile in order to balance the temperature of the target 3D many-core CMPs. To balance the temperature of the tiles of the target 3D many-core CMPs, TCU performs the following steps which are also shown in Algorithm 1 and Figure 6. Algorithm 1 presents the proposed task migration technique and Figure 6 shows, in detail, how to select the optimal coldest tile.

Measuring the Tiles' Temperature and the DRAM Banks' Accesses Percentage Level

The measurement of the temperature of the tiles and the calculations of the DRAM banks accesses' percentage level are prepared as follows:

In the core layer: all tiles read their temperature value from the embedded thermal sensors and then send the thermal information inside control packets to the TCU at the end of each time interval.

***In the cache layer*:** all DRAM banks read the percentage level of their accesses based on embedded performance counters and then send the information inside control packets to the TCU at the end of each time interval. It is noticeable that any DRAM cache bank with a higher percentage level of the accesses has a higher communication level with the tiles. Thus, the higher communication rate results in the DRAM cache banks' getting a higher temperature. Therefore, if one of the DRAM cache banks is the most accessed bank, it means that it is a hotspot DRAM cache bank.

Finding Hotspots and Cold Spots in both Layers

To this end, TCU has gathered statistical information from the core layer and the cache layer. This information includes each tile temperature and their locations in the core layer. In addition, the accesses percentage level of each DRAM bank and their locations in the cache layer are also obtained. In the second step, the TCU analyzes the information it received from step one and then performs the following procedure:

Sorting the tempeature of the tiles from the hottest to the coldest

Filling each entry of the TCU table based on the accesses percentage of the related DRAM bank that is stacked above each tile as it shown in Figure 5.

Dividing the TCU table into four main groups as follows:

$$\left\{\begin{array}{lll} 1 \le i \le 16 & ; \; j = 1 & : \; HOTTEST \\ 17 \le i \le 32 & ; \; j = 2 & : \; MEDHOT \\ 33 \le i \le 48 & ; \; j = 3 & : \; MEDCOLD \\ 49 \le i \le 64 & ; \; j = 3 & : \; COLDEST \end{array}\right\}$$

Where i is the tile number on the sorted TCU table and j is the group number. Finding the hottest tile based on the sorted TCU table.

Finding the Optimal Coldest Tile

In this context, upon obtaining the knowledge of all of the hottest tiles and the coldest tiles in the core layer, the proposed algorithm finds the optimal coldest tile. The TCU analyzes the information in the sorted TCU table and then follows the procedure shown in Figure 6.

Algorithm 1. The proposed task migration technique

Loop:
At the end of each specific time interval:
- The temperature and the location of each tile is determined.
- The accesses' percentage level and the location of each DRAM bank is determined.
- The TCU table is filled by sorting tiles from the hottest to the coldest.
- The accesses' percentage level of each related DRAM bank in the TCU table is tabulated as shown in Figure 5.
- The TCU table is divided into four main parts ($1 \leq j \leq 4$).
- The hottest tile in the core layer is selected.
  Loop:
  The optimal coldest tile is selected as shown in Figure 6.
  End loop;
- Proceed with migration.

End loop;

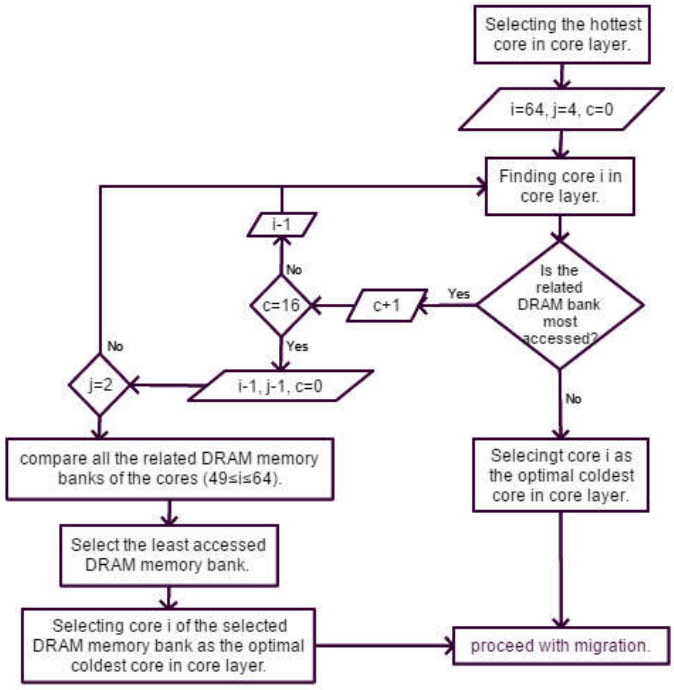

Figure 6. The flowchart of selecting the optimal coldest tile.

As shown in Figure 6, the flowchart of selecting the optimal coldest is presented where *i* is the tile number on the sorted TCU table, *j* is the group number, and *c* is a counter. The TCU starts with the fourth group (*j=4*) which contains the COLDEST tiles ($49 \leq i \leq 64$). TCU then considers the coldest tile (*i=64*) with the accesses percentage level of its related DRAM bank. If the related DRAM bank is not most accessed bank, the TCU will select tile *i* as the optimal coldest tile. Otherwise, TCU checks the second coldest tile (*i=63*) with the accesses percentage level of its related DRAM bank. If the related DRAM bank is not the most accessed DRAM memory bank, TCU selects tile *i* as the optimal coldest tile. Otherwise, TCU will repeat the same procedure with the remaining coldest tiles ($49 \leq i \leq 62$) in the fourth group (*j=4*) in order to find the optimal coldest tile. If the TCU has not found the optimal coldest tile in the fourth group (*j=4*), then it will apply the same procedure within the third group (*j=3*) which contains the MEDCOLD tiles ($33 \leq i \leq 48$). In the rare occurrence where TCU does not find the optimal coldest tile in the third and fourth group, TCU should proceed with the following procedure. It will compare all the related DRAM banks of the 16^{th} coldest tile where (*j=4*) and ($49 \leq i \leq 64$) and then it will select the least accessed DRAM bank amongst them. After that, TCU will choose tile *i* of the selected DRAM bank as the optimal coldest tile which will be migrated with the hottest tile.

Proceeding the migration

Once the TCU finds the optimal coldest tile, it will exchange the thread on the selected optimal coldest tile with the thread on the hottest tile. The proposed task migration mechanism assumes that the whole code and data of the tasks will be exchanged from the hottest tile to the optimal coldest tile. After that, the system should stop the running tasks and proceed to the task migration. At this end, as the TCU table information is updated at the end of each specified time interval that makes it noticeable that the system stays on standby in order to repeat the algorithm in the next time interval. This process is undergone in order to locate the next hottest and optimal coldest tiles to prevent any system failure. Due to the high temperature dependency that exists among vertical adjacent cores and upper adjacent DRAM banks, performing a migration between the hottest and coldest optimal tiles can be very efficient in thermal management. The proposed runtime migration technique has the ability to balance the hotspots in both the core layer and the cache layer at the same time. Therefore, since the system is able to balance hotspots among different cores in different layers, higher overall performance and balanced temperatures are achieved in the 3D many-core CMPs.

Experimental Evaluation

64 cores and 64 DRAM banks of the 3D many-core CMPs architecture with multi-threaded workloads were used to perform the proposed runtime task migration technique.

Platform Setup

To evaluate the proposed method, we used GEM5 [20], a full system simulator to set up the basic system platform. A 64-core architecture which formed a 2D-mesh topology was modeled as shown in Figure 1. Major parameters of the simulation configuration are listed in Table II. Traces from GEM5 were injected to 3D Noxim (Palesi, 2010), a System-C simulator for 3D NoCs modeling. PARSEC benchmarks (Gebhart, 2006), for multi-thread workloads were used in this context. For these benchmarks, one billion instructions were executed for the simlarge input that was set starting from the region of interest (**ROI**). HotSpot (Huang, 2006), version 5.0 was employed as a grid-based thermal modeling tool for 3D temperature estimation. For the experimental evaluation, the maximum temperature limit T_{max} was assumed to be 80□.

Table 2. Specification of the Embedded CMPs Configuration

Component	Description
Number of Cores	64, 8×8 mesh
Core Configuration	Alpha21264, 3GHz, 45nm
Private Cache per each Core	SRAM, 4 way, 32 line, size 32KB per core
On-chip Memory	Baseline: 32MB (64 DRAM banks, each of which has 512KB capacity) Proposed: 32MB (64 DRAM banks, each of which has 512KB capacity) + proposed migration policy

Experimental Results

In this section, the 3D many-core CMP hierarchy were evaluated in two different cases. Firstly, the 3D many-core CMP with DRAM banks in the cache layer stacked on the top of the core layer without any migration policy (Baseline). Secondly, the 3D many-core CMP with a DRAM cache layer stacked on top of the core layer with the proposed migration policy (Proposed).

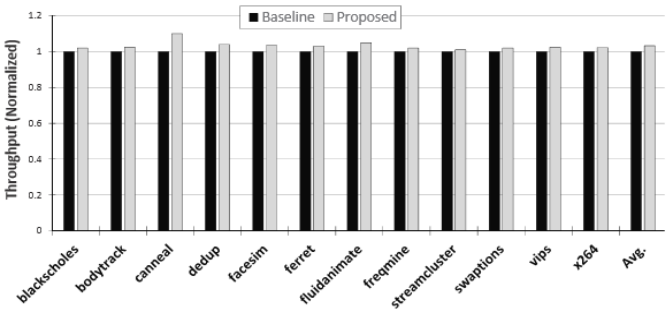

Figure 7. Comparison of throughput results of the PARSEC workloads normalized to the Baseline

Figure 7 shows the results of the normalized throughput for PARSEC workloads, where throughput is the number of executed instructions per second (IPS). As shown in Figure 8, Hybrid-Proposed yields up to 3% throughput improvement when compared with the Baseline. Figure 8 shows the time percentage that the upper layer of the 3D many-core CMP in each case spent, on average, at different temperature points while executing the *canneal* application in order to get a good representation for the cache intensive workloads in PARSEC. As shown in Figure 8, the proposed method ensures that the upper layer DRAM of the 3D many-core CMP are below the maximum temperature of 80℃, while the Baseline architecture spends up to 29% of time above the maximum temperature. As shown in Figure 9, when the *blackscholes* is executed as it is one of the more computation intensive suite in PARSEC benchmarks, the Baseline spent up to 19.5% more of the time above the maximum temperature.

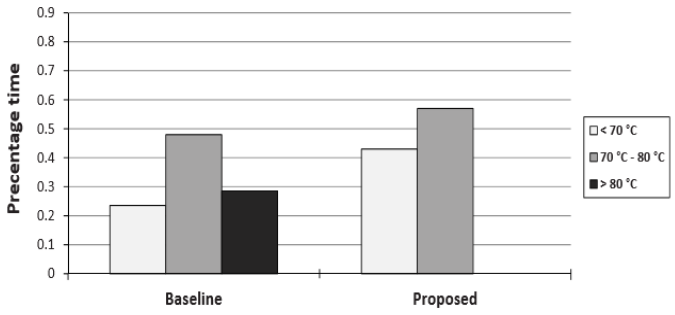

Figure 8. Comparison of the percentage time spent on average by the DRAM layer of the target 3D CMP at different temperature points while executing *canneal*

According to Figure 10, the temperature of the upper cache layer in the proposed architecture is lower than the temperature limit under execution of *canneal*, but in the Baseline architecture, the temperature for cache intensive is above the limit. As shown in this figure, there is on average 13℃ difference between the proposed method and the baseline temperature. Moreover, Figure 11 puts demonstrates that the temperature of the top cache layer in the proposed architecture is lower than the temperature limit under execution of *blackscholes* that is a computation intensive workload, but in the Baseline architecture, the temperature is greater than the limit. As shown in this figure, on average there is a 24℃ difference between the proposed architecture and baseline.

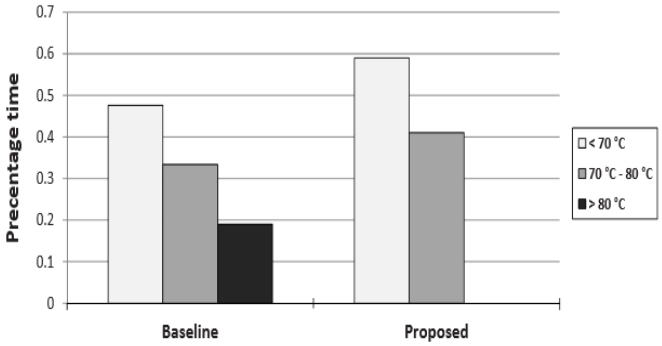

Figure 9. Comparison of the percentage time spent on average by the DRAM layer of the target 3D CMP while executing *blackscholes*

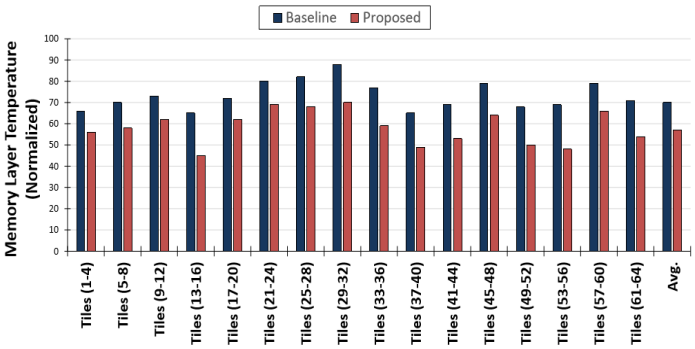

Figure 10. The temperature of the upper layer in *canneal*

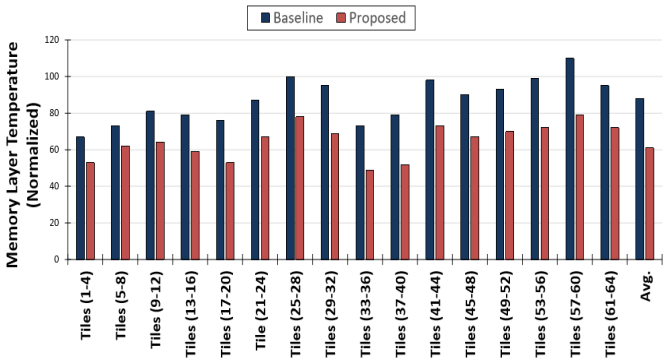

Figure 11. The temperature of the upper layer in *blackscholes*

Figure 12 shows the thermal distribution of the Baseline architecture while executing the memory intensive application *canneal*. As can be seen in this Figure, there are four hotspots in the baseline architecture with a maximum temperature of *110*□ that violates the *80*□ constraint. As shown in Figure 13, the distribution of temperature is more uniform than the baseline architecture and there are not any hotspots in this platform. It can be seen that the difference between the maximum temperature degree in Figure 12 and Figure 13 is *33*□. Moreover, all the workloads of PARSEC were analyzed in this piece and in all cases, the Proposed technique did not violate the maximum temperature constraint. Thus, the

obtained results demonstrated in Figures 12 and 13 show that the proposed task migration technique is efficient in minimizing thermal variance in the target 3D many-core CMP architecture.

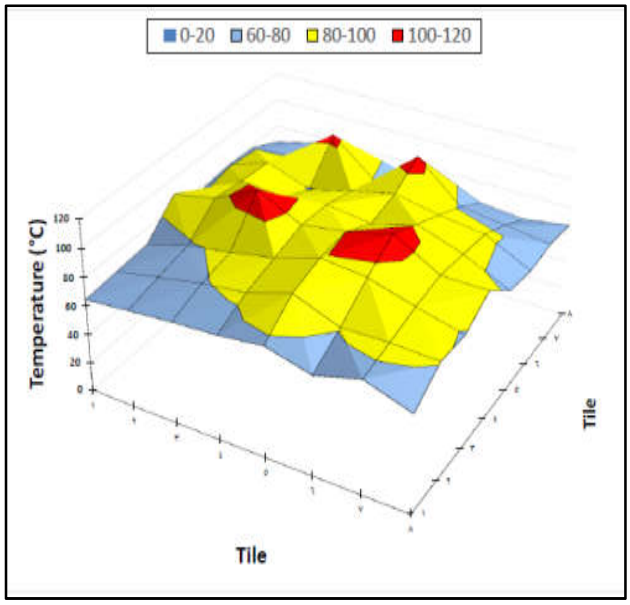

Figure 12. Thermal distribution of the baseline architecture under execution of *canneal*

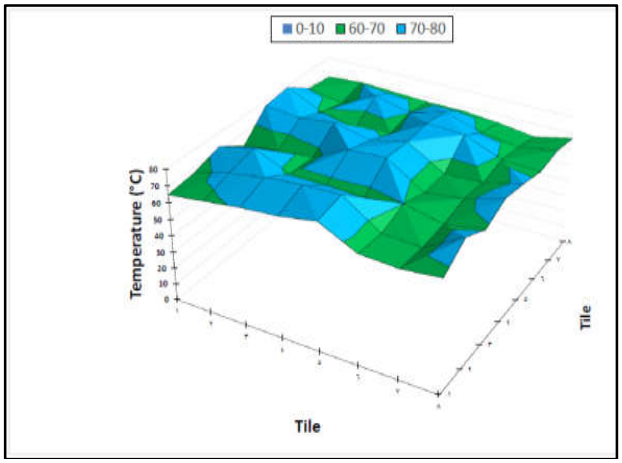

Figure 13. Thermal distribution of the proposed method architecture *canneal*

Conclusion

Applying Runtime Thermal Management (RTM) techniques in the 3D many-core chip multiprocessors (CMPs) has become very important to decreasing the peak temperature without any performance degradation. In this paper, a new runtime task migration technique is presented for 3D many-core CMP in order to balance the temperature gradient and thermal hotspots, and is based on finding the optimal coldest tile that is to migrate with the hottest tile in the core layer. The algorithm continuously analyzes and detects the hottest and the optimal coldest tiles in the core layer with considering the most accessed DRAM banks in cache layer simultaneously. The system selects the optimal coldest tile in the core layer which should not be located under a most accessed DRAM Bank in the cache layer. Since the proposed technique considers hotspots and cold spots in different layers, the obtained results shows a significant reduction of hotspots in the whole 3D many-core CMP architecture. The obtained simulation results indicate up to a 33□ (on average 13□) reduction in the temperature value of the 3D many-core CMP. Moreover, in the simulation results clarified that the proposed technique efficiency was the maximum temperature of cores in the core and cache layers are both less than maximum temperature limit 80□.

REFERENCES

Arora, M. S. Manne , Y. Eckert , I. Paul , N. Jayasena , D. Tullsen , A comparison of core power gating strategies implemented in modern hardware, ACM SIGMET- RICS *Perform. Eval. Rev.* 42 (1) (2014) 559–560.

Asad, A., O. Ozturk, M. Fathy, and M. Jahed-Motlagh, 2015."Exploiting Heterogeneity in Cache Hierarchy in Dark-Silicon 3D Chip Multi-processors," In Digital System Design (DSD), 2015 Euromicro Conference on (pp. 314-321). IEEE, 2015.

Binkert, N., B. Beckmann, G. Black, S. K. Reinhardt, A. Saidi, A. Basu, J. Hestness, D. R. Hower, T. Krishna, and S. Sardashti, 2011. "The gem5 simulator," ACM SIGARCH Computer Architecture News, vol. 39, pp. 1-7.

Chantem, T. Dick, R.P. Hu, X.S. 2011. Temperature-aware scheduling and assignment for hard real-time applications on MPSoCs, IEEE Trans. Very Large Scale Integr. (VLSI) Syst. 19 (10) 1884–1897.

Cheng, H. Y., M. Poremba, N. Shahidi, I. Stalev, M. J. Irwin, M. Kandemir, J. Sampson, and Y. Xie. "EECache: Exploiting design choices in energy-efficient last-level caches for chip multiprocessors." In Proceedings of the 2014 international symposium on Low power electronics and design, pp. 303-306. ACM, 2014.

Cheng, Y., Zhang, L., Han, Y., and Li, X. (2013). Thermal-constrained task allocation for interconnect energy reduction in 3-D homogeneous MPSoCs. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 21(2), 239-249.

Coskun, A. K. J. L. Ayala, D. Atienza, T. Simunic Rosing, and Y. Leblebici. "Dynamic thermal management in 3D multicore architectures." In Proceedings of the Conference on Design, Automation and Test in Europe, pp. 1410-1415. European Design and Automation Association, 2009.

Cui, Y., Zhang, W., Chaturvedi, V., Liu, W., and He, B. 2016. Thermal-Aware Task Scheduling for 3D-Network-on-Chip: A Bottom to Top Scheme. Journal of Circuits, Systems and Computers, 25(01), 1640003.

Donald, J., M. Martonosi, 2006. Techniques for multicore thermal management: Classification and new exploration, ACM SIGARCH Comput. Archit. News 34 (2), 78–88.

Gebhart, S. M., Gebhart, Mark, Joel Hestness, Ehsan Fatehi, Paul Gratz, and Stephen W. Keckler. 2009. "Running PARSEC 2.1 on M5." University of Texas at Austin, Department of Computer Science, Technical Report.

Hanumaiah, V. and S. Vrudhula. 2014. "Energy-efficient operation of multicore processors by DVFS, task migration, and active cooling." IEEE Transactions on Computers 63, no. 2: 349-360.

Hassanpour, N., Hessabi, S., and Hamedani, P. K. 2013. Temperature control in three-network on chips using task

migration. IET Computers and Digital Techniques, 7(6), 274-281.

Henkel, J. H. Khdr , S. Pagani, M. Shafique. 2015. New trends in dark silicon, in: Proceedings of the ACM/EDAC/IEEE Design Automation Conference (DAC), pp. 1–6.

Henkel, J. H. Khdr, S. Pagani, M. Shafique, 2015. New trends in dark silicon, in: Proceedings of the ACM/EDAC/IEEE Design Automation Conference (DAC), pp. 1–6.

Heo, S., K. Barr, K. Asanovi ´c, Reducing power density through activity migration, in: Proceedings of the International Symposium on Low Power Electronics and Design (ISPLED), 2003, pp. 217–222.

Huang, W., Ghosh, S., Velusamy, S., Sankaranarayanan, K., Skadron, K., and Stan, M. R. 2006. HotSpot: A compact thermal modeling methodology for early-stage VLSI design. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 14(5), 501-513.

Kadjo, D., H. Kim, P. Gratz, J. Hu, and R. Ayoub. 2013. "Power gating with block migration in chipmultiprocessor last-level caches." In Computer Design (ICCD), IEEE 31st International Conference on, pp. 93-99. IEEE, 2013.

Kadjo, D., Kim, H., Gratz, P., Hu, J., and Ayoub, R. (2013, October). Power gating with block migration in chip-multiprocessor last-level caches. In Computer Design (ICCD), 2013 IEEE 31st International Conference on (pp. 93-99). IEEE.

Kang, K., J. Kim, S. Yoo, and C. M. Kyung. 2011. "Runtime power management of 3-D multi-core architectures under peak power and temperature constraints." IEEE Transactions on ComputerAided Design of Integrated Circuits and Systems 30, no. 6: 905-918.

Kang, U., Chung, H.J., S. Heo , D.H. Park , H. Lee , J.H. Kim , *et al*. 2010. 8 gb 3-d ddr3 dram using through-silicon-via technology, IEEE J. Solid-State Circuits 45 (1) 111–119.

Kim, W. M.S. Gupta, G.-Y. Wei , D. 2008. Brooks, System level analysis of fast, per–core DVFS using on-chip switching regulators, in: Proceedings of the International Symposium on High Performance Computer Architecture (HPCA), 2008, pp. 123–134.

Kumar, A. S. Li , P. Li-Shiuan, N.K. Jha, System-level dynamic thermal management for high-performance microprocessors, IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst. 27 (1) (2008) 96–108.

Lee, B. Ipek, E. O. Mutlu, D. Burger, 2009. Architecting phase change memory as a scalable DRAM alternative, ACM SIGARCH Comput. Archit. News 37 (3) 2–13.

Li, J., Qiu, M., Niu, J. W., Yang, L. T., Zhu, Y., and Ming, Z. 2013. Thermal-aware task scheduling in 3D chip multiprocessor with real-time constrained workloads. ACM Transactions on Embedded Computing Systems (TECS), 12(2), 24.

Li, Q. Y. He, J. Li, L. Shi, Y. Chen, C.J. Xue, Compiler-assisted refresh minimization for volatile STT-RAM cache, IEEE Trans. Comput. 64 (8) (2015) 2169–2181.

Li, Q., J. Li, L. Shi, M. Zhao, C.J. Xue , Y. He, Compiler-assisted STT-RAM-based hybrid cache for energy efficient embedded systems, IEEE Trans. Very Large Scale Integr. (VLSI) Syst., 22 (8) (2014) 829–1840.

Liu, Z., Tan, S. X. D., Huang, X., and Wang, H. 2015. Task migrations for distributed thermal management considering transient effects. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 23(2), 397-401.

Liu, Z., Xu, T., Tan, S. X. D., and Wang, H. (2013, January). Dynamic thermal management for multi-core microprocessors considering transient thermal effects. In Design Automation Conference (ASP-DAC), 2013 18th Asia and South Pacific (pp. 473-478). IEEE.

March, J. L., Sahuquillo, J., Petit, S., Hassan, H., and Duato, J. (2013). Power□aware scheduling with effective task migration for real□time multicore embedded systems. Concurrency and Computation: Practice and Experience, 25(14), 1987-2001.

Meng, J., A.K. Coskun, Analysis and runtime management of 3D systems with stacked DRAM for boosting energy efficiency, in: Proceedings of the De- sign, Automation and Test in Europe Conference and Exhibition (DATE), 2012, pp. 611–616.

Murali, S. A. Mutapcic, D. Atienza, R. Gupta, S. Boyd, and G. De Micheli, "Temperature-aware processor frequency assignment for MPSoCs using convex optimization," in Proc. CODES+ISSS, Sep. 2007, pp. 111–116.

Murali, S. A. Mutapcic, D. Atienza, R. Gupta, S. Boyd, L. Benini, and G. De Micheli. 2008. "Temperature control of high-performance multi-core platforms using convex optimization." In Design, Automation and Test in Europe, 2008. DATE'08, pp. 110-115. IEEE.

Palesi, M., S. Kumar, and D. Patti, 2010. "Noxim: Network-on-chip simulator" . http://noxim.sourceforge.net.

Rasquinha, M., Choudhary, D. Chatterjee, S. Mukhopadhyay, S. Yalamanchili, S. 2010. An energy efficient cache design using Spin Torque Transfer (STT) RAM, in: Proceedings of the 16th ACM/IEEE International Symposium on Low Power Electronics and Design (ISLPED), 2010, pp. 389–394.

Rotem, E. A. Mendelson, R. Ginosar , U. Weiser, Multiple clock and voltage do- mains for chip multi processors, in: Proceedings of the 42nd Annual IEEE/ACM International Symposium on Microarchitecture (Micro), 2009, pp. 459–468.

Srinivasan, S. 2012. Functional Test Pattern Generation for Maximizing Temperature in 2D and 3D Integrated Circuits.

Sun, G., X. Dong, Y. Xie, J. Li , Y. Chen, 2009. Novel architecture of the 3D stacked MRAM L2 cache for CMPs, in: Proceedings of the 15th International Symposium on High Performance Computer Architecture (HPCA), pp. 239–249.

Syu, S. M., Shao, Y.H., Lin, I. C. 2013. High-endurance hybrid cache design in CMP architecture with cache partitioning and access-aware policy, in: Proceedings of the 23rd ACM International Conference on Great Lakes Symposium on VLSI (GL SVL SI), pp. 19–24.

Tajik, H., H. Homayoun, N. Dutt, 2013. VAWOM: Temperature and process variation aware wearout management in 3D multicore architecture, in: Proceedings of the 50th Annual Design Automation Conference (DAC), 2013, pp. 1–8.

Wang, Z., D.A. Jimenez, C. Xu, G. Sun, Y. Xie, 2014. Adaptive Placement and Migration Policy for an STT-RAM-Based Hybrid Cache, in: Proceedings of the High Performance Computer Architecture (HPCA), pp. 13–24.

Zhan, J., J. Ouyang, F. Ge, J. Zhao, and Y. Xie. "DimNoC: A dim silicon approach towards power-efficient on-chip network." In Design Automation Conference (DAC), 2015 52nd ACM/EDAC/IEEE, pp. 1-6. IEEE, 2015.

Zhan, J., Y. Xie, and G. Sun. "NoC-Sprinting: Interconnect for fine-grained sprinting in the dark silicon era." In Design Automation Conference (DAC), 2014 51st ACM/EDAC/IEEE, pp. 1-6. IEEE, 2014.

Zhao, D., H. Homayoun, A.V. 2013. Veidenbaum, Temperature aware thread migration in 3D architecture with stacked DRAM, in: Proceedings of the ISQED, 2013, pp. 80–87.

Zhao, D., Homayoun, H., and Veidenbaum, A. V. 2013, March). Temperature aware thread migration in 3D architecture with stacked DRAM. In Quality Electronic Design (ISQED), 2013 14th International Symposium on (pp. 80-87). IEEE.

****