Can Dark Silicon Be Exploited to Prolong System Lifetime?

Mohammad-Hashem Haghbayan[1], Antonio Miele[3], Amir M. Rahmani[1], Pasi Liljeberg[1],
Axel Jantsch[4], Cristiana Bolchini[3] and Hannu Tenhunen[1,2]
[1]Department of Information Technology, University of Turku, Finland
[2]Department of Industrial and Medical Electronics, KTH Royal Institute of Technology, Sweden
[3]Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, Italy
[4]Institute of Computer Technology, TU Wien, Austria
E-mail: mohhag@utu.fi, antonio.miele@polimi.it, amirah@utu.fi, pakrli@utu.fi,
axel.jantsch@tuwien.ac.at, cristiana.bolchini@polimi.it, hannu@kth.se

***Abstract*—With the break down of the Dennard scaling, we entered the Dark silicon era, where the available power budget is no more able to feed all the cores available within the same chip at full throttle. At the same time, the extreme downscaling of CMOS technologies has caused an acceleration in device aging and wear-out processes. In the dark silicon era, runtime resource management in many-core systems becomes more challenging as many aspects have to be considered all together, such as power capping, dynamic applications mapping, performance improvement and also reliability management. In this paper, we claim that dark silicon can be exploited for reliability purposes by efficiently managing system resources (both cores and power) in order to prolong the system lifetime while achieving the same level of performance.**

***Index Terms*—Dark silicon, many-core systems, lifetime reliability, aging, dynamic power management, runtime mapping**

I. Introduction

Technology scaling has brought massive transistor integration per unit area thus making it possible to fabricate hundreds of cores within the same chip, dubbed as many-core system. However, with the end of Dennard scaling, supply voltage has not followed the same exponential scaling trend experienced with transistors. As a consequence, physical limits imposed by device packaging and cooling technology on peak power consumption and peak power density have made it impossible to power-on the entire chip at the same time, leading to the *dark silicon* problem [1]. In practice, to avoid high temperatures that would damage transistor junctures, a chip is generally constrained by a given Thermal Design Power (TDP), or recently by the more advanced Thermal Safe Power (TSP, [2]). In particular, they allow to activate only a subset of the processing cores at the nominal voltage/frequency level, i.e. at maximum performance level, while the rest of the resources must remain power-gated idle mode. According to projections of the International Technology Roadmap for Semiconductors (ITRS) in 2013 [2], for CPU design at 22nm the percentage of dark silicon is around 50% while for 8nm it will increase to 70%; this represents a critical issue for near future many-core systems.

The adoption of a TDP (or TSP) only partially solves the issues related to the increased power densities, and the resulting high operating temperatures within the device. In fact, even if TDP avoids excessive temperature peaks, the overall temperature profiles that characterize modern devices are considerably higher than in the past. As discussed in the ITRS reports in 2011 [3], such high temperatures, combined with the extreme downscaling of CMOS technologies, have caused an acceleration in device aging and wear-out processes. As a matter of fact, modern circuits are more susceptible to phenomenon such as Electromigration or Time Dependent Dielectric Breakdown, that lead to circuit degradation causing delay errors and, eventually, device breakdowns. Past studies [4] have shown that many failure mechanisms are exponentially dependent on temperature, and a $10-15°$C difference in operating temperature may result in a $2\times$ difference in the overall lifespan of a device. Thus, we are currently experiencing a dramatic decrease of lifetime in modern digital system; therefore, reliability is becoming a primary design driver in current and near-future systems.

In the past years, researchers (e.g. [5]–[9]) have proposed system-level strategies for slowing down the aging process. The main idea is to balance the utilization of processing cores and use dynamic voltage and frequency scaling (DVFS) to keep the operating temperature and the accumulated stress under control over the system's service time. However, the management of a many-core system is a complex activity where several matters need to be considered:

- Dynamics of the running workload with applications entering and leaving the system with an unknown trend.
- Applications may be composed by interdependent tasks whose distribution will considerably affect the overall performance due to the generated traffic on the communication infrastructure.
- Dynamic power consumption and the imposed power budget, especially considering TDP.

For these reasons, many-core systems are generally provided with an advanced Runtime Resource Management (RRM)

layer managing applications' mapping and power distribution. In this scenario, the straightforward integration of existing approaches is not effective, since they only consider a part of the complex picture, and often, they have partially contradicting objectives with the RRM policies viz. to enhance power-performance characteristics versus to enhance balanced allocation. In [6] and [9], the management techniques can be only applied to single-thread applications. The approaches presented in [5], [7], [10] only adopt simplified mapping policies as their focus is on shared-memory architectures. In [8], power management is not considered in workload assignment process. For these reasons, we claim that, to enhance the many-core systems' lifetime within the dark silicon context, it is necessary to act in an essentially different way: *to consolidate runtime resource management and reliability control*. More precisely, nominal policies integrated in the runtime resource management layer must be enhanced to provide reliability-awareness.

Indeed, dark silicon may represent new opportunities that have not been fully explored yet. In fact, the abundance of cores and the infeasibility to use all of them at the same time provide a unique opportunity for the runtime management unit to spread the utilization stress among the processing units to prolong the system lifetime. In the following, we discuss the opportunities given by dark silicon for lifetime improvement in many-core systems by presenting empirical evidence derived from an extensive set of experiments. Moreover, we elaborate on the challenges related to the definition of reliability-aware runtime resource management strategies for the considered architecture under dark silicon scenario. Our experiments demonstrate a reliability-aware naive proof-of-concept runtime resource management approach being capable of improving the lifetime of the system up to 41% w.r.t. its nominal counterpart. It should be noted that further improvements can also be achieved by using more advanced techniques.

II. Reliability Model

Reliability model defined in the industrial standards [11] generally adopts statistical distributions to characterize the temporal failure density in the devices, such as the Weibull or lognormal ones. More precisely, the reliability function is defined with the Weibull distribution as [4]:

$$R(t) = e^{-\left(\frac{t}{\alpha}\right)^{\beta}} \tag{1}$$

where α is the scale parameter and β the shape parameter. Moreover, α is defined according to a given failure model. For instance, when considering the Electromigration (EM) failure mechanism, α is:

$$\alpha = \frac{A_{EM}(J - J_{\text{crit}})^{-n} e^{\frac{E_{a_{EM}}}{kT}}}{\Gamma\left(1 + \frac{1}{\beta}\right)} \tag{2}$$

where A_{EM} is a material-dependent constant, J is the current density, J_{crit} is the critical current density activating the phenomenon, n is empirically determined constant, $E_{a_{EM}}$ is the activation energy, K is Boltzman's constant, and T is the constant worst-case temperature.

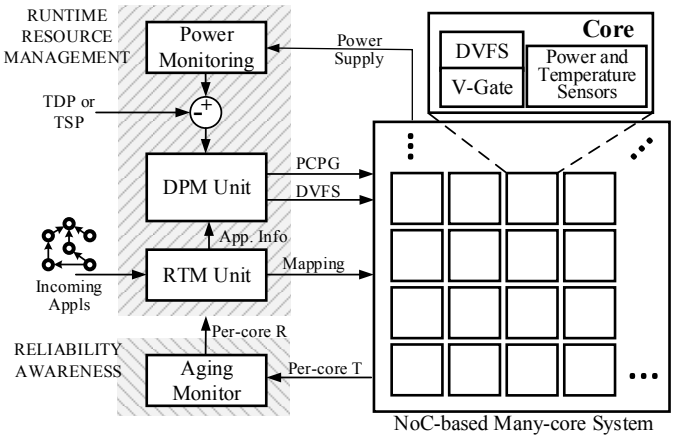


Figure 1: System architecture and companion RRM layer.

This model cannot be directly used in system-level reliability analysis, due to the fact that it only considers fixed worst-case temperature and utilization, and is not suitable to handle changes in the operating conditions. For this reason, in [4] the reliability function has been re-formulated as:

$$R(t) = e^{-\left(\sum_{j=1}^{i} \frac{\tau_j}{\alpha(T_j)}\right)^{\beta}} \tag{3}$$

where τ_j represents the duration of each period of time having a constant steady-state temperature T_j until time t (i.e., $t = \sum_{j=1}^{i} \tau_j$). Then, the expected lifetime of the system, measured in terms of the Mean Time To Failure (MTTF), is measured as the area beneath the curve:

$$MTTF = \int_0^{\infty} R(t)dt \tag{4}$$

Finally, since the considered architecture has multiple units, we need to combine the estimated lifetimes of the various units. As in [6], we approximate the system's MTTF to the minimum of the units' MTTFs.

III. Many-core System Architecture

Figure 1 shows the target architecture. The hardware platform is the classical NoC-based many-core architecture containing homogeneous processing cores. This system is suitable for executing data-intensive and highly-parallel applications, organized in a set of inter-dependent elaboration tasks. Tasks are mapped on the cores and communications among tasks are based on a message passing protocol over the NoC.

Many-core systems frequently experience a highly-variable workload, consisting of applications with different resource and power requirements, arriving with an unknown trend. Moreover, they are constrained by a given TDP or TSP. Therefore, the activities of the system are dynamically coordinated by a Runtime Resource Management (RRM) layer [12], containing a Runtime Mapping (RTM) unit and a Dynamic Power Management (DPM) unit.

The RTM unit is in charge of dispatching the incoming applications on the cores. The prerequisites are the presence of a set of idle units and a minimum power budget for the application execution; this last condition is ensured by the DPM unit. To dominate the complexity of this phase, the RTM acts in two steps. First, the region selection is performed to find a set of neighboring idle cores to be reserved for the new application. Then, application tasks are actually mapped into the selected region aiming at reducing communication latencies.

The DPM unit distributes the available power budget among the cores and informs the RTM unit about the feasibility of the mapping of the new application. DPM unit may use two alternative strategies. The first one considers only the per-core power gating (PCPG) knob and activates only a subset of the cores at maximum performance based on the TDP/TSP and the actual power request of the running applications. The second one performs a more accurate power management by using also DVFS so that it is possible to trade-off the number of active cores for the provided per-core performance level according to applications' Quality of Service (QoS) requirements.

In the analysis, we have considered state-of-the-art strategies for the various phases; in particular, in the two-step runtime mapping we considered MapPro and CoNA [2], and for the dynamic power management both PAM [12], which only uses PCPG, and MOC [12] using also DVFS.

IV. Reliability Awareness in Runtime Resource Management

In order to explore the opportunities offered by dark silicon, we extend the framework discussed in the previous section to enable lifetime reliability awareness by introducing a reliability monitor (similarly to [6], [9]), as shown in Figure 1, based on the temperature sensing and implementing the aging model discussed in the background section. Such novel reliability monitor is then exploited as an additional input for the RRM layer to drive decisions also according to the aging conditions of the various cores within the architecture. The next subsections discuss how each step can be extended, presenting the rationale and benefits at the basis of these decisions.

A. Reliability-aware Region Selection

The rationale at the basis of the nominal region selection is to identify a possibly-rectangular region that maximizes the availability of idle cores and fulfills a network contiguity metric in order to minimize network traffic. To introduce reliability awareness, we may exploit the fact that dark silicon forces a large number of nodes to stay idle, as a degree of freedom in region selection. More precisely, at this step, we may work at a coarse-grained level by selecting the areas of the grid that will be active and the ones that will be dark according to the aging status of the cores, thus deciding the distribution of the thermal stress.

To evaluate this opportunity, we defined a naive reliability-aware approach as an extension of MapPro [2], one of the most recent region selection strategies. The strategy searches the most suitable vacant region that is close-by to the other mapped applications. MapPro exploits two weights for each node of the network:

- The **Vicinity Counter** (VC) is a metric to quantify the occupied nodes in each square-shaped region.
- The **Distance Factor** (D) measures the proportional distance of the node from the other occupied nodes.

Two types of congestion can affect the execution time of a multi-task application, i.e., internal congestion caused by intra packet exchange between application's tasks, and external congestion caused by interference of other applications' packets. Based on the definition of VC and D values, selecting a region with a higher VC value for application mapping provides more vacant area in a square shaped region which causes the application's tasks to be closer to each other, thus reducing the internal congestion. Furthermore, choosing a lower D value denotes the selected region is closer to other running applications which keeps contiguity of mapped applications, and, thus, reduces the external congestion. When a new application must be mapped, MapPro determines the size of minimum squared region to be allocated for its execution on the basis of the application size. Then, it identifies all the suitable square regions, and selects the one maximizing the VC value, and, secondarily, minimizing D value.

To make this algorithm sensitive to the cores' aging, the enhanced strategy acts as follows:

1) It performs a first selection of a subset of candidate regions having VC and D values within a given percentage threshold from the best values.
2) It computes, for each region, a reliability metric $R_{min}(t)$, being the minimum reliability level of the included cores.
3) Finally, it selects within the subset the region that maximizes $R_{min}(t)$, maximizes VC and minimizes D values, in this order.

As a result, if there are two squares with the VC and D values that are not much different, the approach selects the area with "younger" cores. Do note that the more dark cores are present on the system, the higher the number of square shaped regions with almost the same VC and D values. Therefore, the chance of finding different regions with the same network-constrained metrics increases, and this can be useful to better balance reliability in the long-term.

To exemplify the idea, let us consider the scenario depicted in the top part of Figure 2(a): there are four running applications and we aim at starting a newly arrived application 5. Therefore, In case of traditional strategy, there will be various regions optimizing the VC and D metrics, to place application 5 side by side to the other ones in execution and, therefore, reduce the dispersion of utilized cores; among them MapPro performs the selection by chance, as shown in the left-bottom part of Figure 2(a). This choice may not be optimal in terms of the aging affecting the architecture; in fact, as shown by the reliability values annotated on the grid, we continue using a group of cores that have already been subjected to a

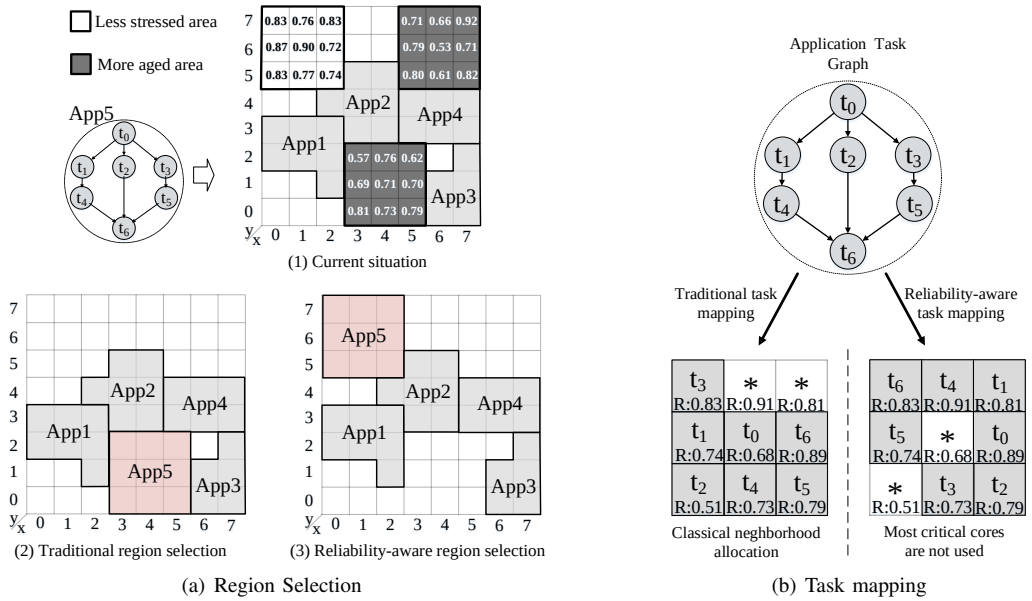


Figure 2: Example of different traditional and reliability-aware choices in region selection and task mapping.

considerable amount of stress. On the contrary, in the case the region selection strategy is able to take into account also an aging metric, it has the opportunity to perform a reliability-aware choice by using the younger cores, as shown in the right-bottom part of Figure 2(a). In this way, on the long period the enhanced strategy will be able to balance the aging among the architecture grid and to obtain an overall prolonging of the system lifetime.

B. Reliability-aware Task Mapping

The metric that is generally adopted in the intra-region mapping is contiguity in the placement of interdependent tasks, to reduce communication delays. However, since the region frequently contains a number of cores larger than the smallest one necessary to execute the application, we may execute the task mapping algorithm to avoid the use of elder cores.

Therefore, we extended a task mapping strategy called CoNA [2], to introduce reliability awareness. The main idea for this naive reliability-aware strategy is to prevent the usage of the most aged cores by tagging as critical all the cores violating the following condition:

$$\frac{R_i(t) - R_{avg}(t)}{R_{avg}(t)} > \%R_{thr} \tag{5}$$

where $R_i(t)$ is the current reliability of core i, $R_{avg}(t)$ is the average architecture reliability and $\%R_{thr}$ is a given threshold. Then, CoNA tries to map initially without using such cores and, in case of failure, by using (a part of) them.

If we consider the application to be mapped into the selected region shown in the top part of Figure 2(b), we have a few possible choices to distribute the tasks according to the connections and by exploiting the two spare cores w.r.t. the number of tasks obtained by the selection of a rectangular region. The traditional strategy considers just only the connectivity to compact as much as possible tasks and reduce communication times (as shown in the left-bottom part of Figure 2(b). However, this choice does not consider the presence of cores that have suffered from a considerable stress and are consequently aged considerably. At the opposite, a reliability-aware strategy tries to put such cores in an idle state while considering the traditional connectivity metric as well. In that way the application achieves approximately same performance of the traditional approach, while the most aged cores are restored for a while (shown in right-bottom part of Figure 2(b)).

It should also be noted that the aging status of a specific core is also affected by the heating caused by neighboring cores' activity. Therefore, discarding one core for mapping might not necessarily stop aging process completely. From this perspective, the previous region selection strategy will have a more beneficial outcome since it works at a coarse-grained level by not using entire regions containing excessively-aged cores.

C. Reliability-aware Power Management

When considering the power management, only the DVFS-based strategy may be extended with reliability awareness. Acting on PCPG would result in a mapping approach, since the node can only be switched on or off.

MOC [12] performs power management by monitoring the overall current power consumption and other network characteristics, such as router congestion and applications injection rate, and actuates in a feedback loop on the per-core DVFS knobs. Since power consumption has direct effect on temperature, which is the main cause in the cores' aging, we consider the current reliability of the cores in the power management algorithm and limit the maximum frequency of the more stressed areas proportional to other areas.

To implement this idea, we enhanced MOC to restrict the maximum frequency of the most aged areas. This frequency limitation should not be too greedy not to have a highly negative impact on the application performance. Then, the amount of power saved by this limitation can be used to accelerate other applications running on other younger areas that cause the overall throughput of the system approximately unchanged comparing to reliability-agnostic approach.

RTM unit is also aware of such areas and does not map performance-demanding applications on them unless there is not any vacant core in younger regions. In such cases, the frequency restriction will be discarded and the application can be run at high frequency. Even though cutting frequency might have a minor contribution to power reduction in the local area, in the long term it helps to balance the stress on the chip.

V. Experimental Results

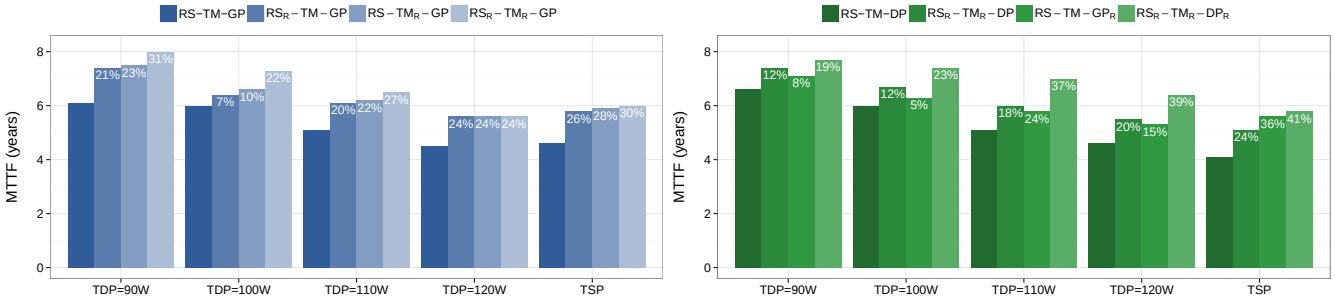
We evaluated the presented ideas by means of the SystemC simulator used for analyzing the nominal approaches [12]. We defined a 12×12 many-core architecture with the physical and architectural characterization of the Niagara2-like core implemented with a $16nm$ technology (the overall chip area is $138mm^2$). We considered four different TDP values, $90W$, $100W$, $110W$, and $120W$, and also TSP with a maximum temperature of 80°C (as in [2]). Moreover, we considered the EM phenomenon, characterizing the constants according to [11]; A_{EM} was set to have a 10-year lifetime for a single core independently implemented and working at a steady-state temperature of 60°C. Finally, we defined a dynamic workload as in [12] in terms of several types of multi-task applications randomly entering and exiting the system at runtime. Two types of applications are modeled, i.e., non-realtime and soft realtime, where non-realtime applications have lower priority than soft realtime. Applications entering sequence is kept fixed for all experiments for the sake of comparison.

We performed an evaluation of each single reliability-aware runtime strategy against the nominal RRM layer, used as a baseline. Moreover, we incrementally combined the enhanced mechanisms per pairs and we finally run the full-fledged reliability-aware approach. We performed this analysis by considering both the PCPG power management strategy and the DVFS-based one. The outcomes of such an analysis are shown in Figure 3, where, for the sake of clarity, only the most relevant results are reported. The naming convention specifying which combination of approaches has been used in each single run is the following: `RS`, `TM`, `GP` and `DP` represent the nominal strategies for region selection, task mapping, PCPG-based power management and DVFS-based power management, respectively, while the reliability-aware counterparts are identified with a subscript `R`, i.e., $\texttt{RS}_\texttt{R}$, $\texttt{TM}_\texttt{R}$ and $\texttt{DP}_\texttt{R}$.

A first straightforward evidence is that dark silicon is intrinsically beneficial for lifetime improvement. In fact, an increase in the TDP value leads in a degradation of the overall lifetime, also in case of reliability-agnostic approaches. This is due to the fact that a higher TDP will lead to higher temperatures but at the same time a higher computational efficiency, thus reaching higher throughput (in terms of the number of applications completed per hour). It should be noted that while using TSP, the maximum power bound dynamically changes according to the current situation of tasks running on the system. In our experiment TSP becomes more than 120W which results in reliability degradation more than the other scenarios. Do note also that in the overall analysis we cannot directly compare absolute MTTF values among different scenarios (characterized by different TDP/TSP values or adoption of PCPG/DVFS mechanisms), since they are characterized by different performance results (as discussed in [12]); therefore, the only proper comparison would be to consider performance and MTTF values obtained in each scenario, together. Moreover, in the same working scenario, we experimentally confirmed that every combination of reliability-aware strategies obtained approximately the same performance as the reliability-agnostic baseline, with an error lower than 1% (results are not reported for the sake of space).

When considering the PCPG scenario, we may conclude that both the two independent reliability-aware steps of the mapping have a beneficial effect, for instance, up to 21% and 23% for $90W$, respectively, and their combination provides a further MTTF improvement up to 31%. However, in some situations, such as $120W$, their combination is not beneficial due to the fact that in the $120W$ scenario, the dark silicon amount is quite limited. When using DVFS, we may have further improvement in terms of lifetime compared to the nominal baseline, especially in the scenarios where TDP is set to 120W as well as where TSP is used (where the power consumption is even higher). Moreover, when using the full-fledged approach in the scenario with DVFS actuation $\texttt{RS}_\texttt{R}\texttt{-TM}_\texttt{R}\texttt{-DP}_\texttt{R}$, we notice that the percentage trend of lifetime improvement is increasing w.r.t. the increasing TDP budget, while we experienced an opposite trend in the scenario with PCPG actuation ($\texttt{RS}_\texttt{R}\texttt{-TM}_\texttt{R}\texttt{-GP}$). This is related to the fact that the actuation on DVFS allows to further keep the thermal stress under control.

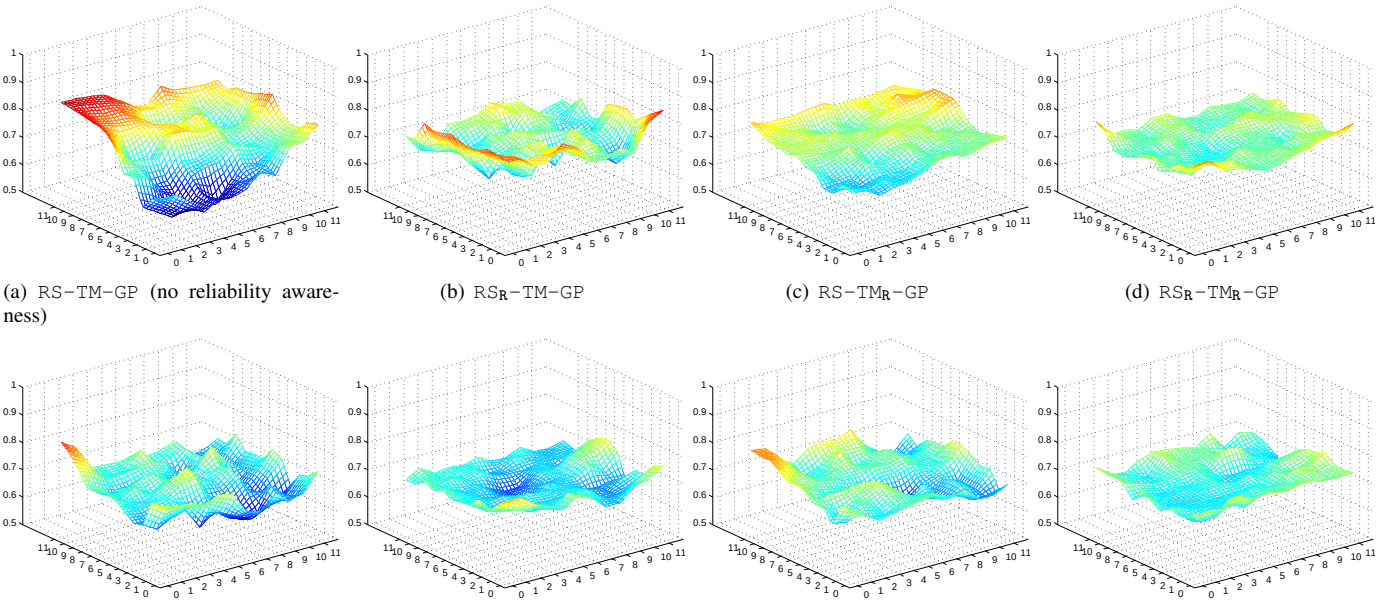
In order to better analyze the effects of dynamic resource management on reliability, we present in Figure 4 the reliability status of the various cores after 5 years of activity under the influence of the various reliability-aware and agnostic control strategies. Figures 4(a) and 4(e) confirm that reliability-agnostic DVFS is intrinsically able to better mitigate the aging on the system than PCPG. This comes from the fact that, in DVFS process, a voltage/frequency pair decreases together per steps which results in cubic reduction in dynamic power and direct reduction in static power. Such a high power reduction alongside with the increase in parallelization result in a more distributed on-chip power density and, consequently, mitigation of thermal hotspots. Then, other plots confirm that the various single approaches for mapping and DVFS control are able to better exploit dark silicon and to improve reliability, while the combination of all the approaches offers best experimental results. In the last experimental session, we also compared $\texttt{RS}_\texttt{R}\texttt{-TM}_\texttt{R}\texttt{-DP}_\texttt{R}$ against some existing system-



(a) Considering PCPG power control strategy

(b) Considering DVFS power control strategy

Figure 3: MTTF obtained with the different strategies.



(a) RS−TM−GP (no reliability awareness)

(b) $\mathrm{RS_R}$−TM−GP

(c) RS−$\mathrm{TM_R}$−GP

(d) $\mathrm{RS_R}$−$\mathrm{TM_R}$−GP

(e) RS−TM−DP (no reliability awareness)

(f) $\mathrm{RS_R}$−$\mathrm{TM_R}$−DP

(g) RS−TM−$\mathrm{DP_R}$

(h) $\mathrm{RS_R}$−$\mathrm{TM_R}$−$\mathrm{DP_R}$

Figure 4: Lifetime reliability of the cores after 5 years with different strategies when considering TDP = $90W$.

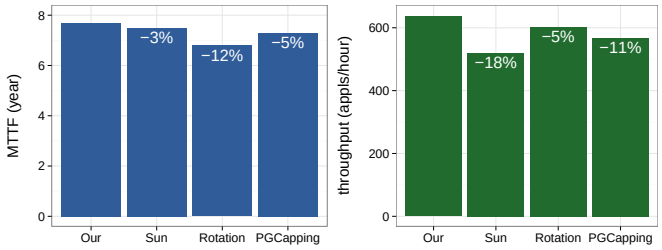
level reliability-driver approaches: *Sun* [8], *rotation* [9], and *PGCapping* [5]. These approaches have been integrated in the considered nominal RRM layer. Figure 5 presents the achieved results considering a TDP of $90W$. Since the various approaches obtain different performance, to have a fair comparison we report also the related throughput. From the results, it is possible to conclude that the proposed approach outperforms the other ones both in terms of MTTF and throughput. In particular, even if *Sun* and *PGCapping* obtain MTTFs very close to the one achieved by the proposed approach (around -5%), they pay a considerable penalty on the performance (up to -18%). At the opposite, *Rotation* presents a limited performance overhead (-5%), and as a counterpart, a shortened MTTF (-12%). This confirms our initial claim that it is necessary to tackle reliability issues within each module of the nominal RRM layer in a holistic way, instead of re-adapting state-of-the-art system-level policies without taking into account the overall picture at the same time, as done in the considered past approaches. Nevertheless, it is worth remarking that we proposed naive policies as a proof-of-concept for demonstrating our claim. We believe that it is possible to define more advanced techniques in the same direction allowing to achieve larger improvements.

VI. Conclusions

This paper provided an idea of how dark silicon may represent an opportunity to increase the lifetime of a many-core system by introducing reliability awareness in runtime resource management. Our naive proof-of-concept approach succeeded to achieve a 20% to 40% lifetime improvement, and our vision is that much more enhancements can be pursued if more advanced techniques are invoked.

References

[1] H. Esmaeilzadeh *et al.*, "Dark Silicon and the End of Multicore Scaling," *IEEE Micro*, vol. 32, no. 3, 2012.



(a) Lifetime (b) Performance

Figure 5: Comparison w.r.t. state-of-the-art approaches.

[2] A. Rahmani, P. Liljeberg, A. Hemani, A. Jantsch, and H. Tenhunen, *The Dark Side of Silicon*, 1st ed. Springer, 2016.

[3] ITRS, "International Technology Roadmap for Semiconductors," http://www.itrs2.net/.

[4] Y. Xiang, T. Chantem, R. Dick, X. Hu, and L. Shang, "System-level reliability modeling for MPSoCs," in *Proc. Conf. on Hardware/Software Codesign and System Synthesis (CODES)*, 2010, pp. 297–306.

[5] K. Ma and X. Wang, "PGCapping: Exploiting Power Gating for Power Capping and Core Lifetime Balancing in CMPs," in *Proc. Int. Conf. on Parallel Architectures and Compilation Techniques (PACT)*, 2012, pp. 13–22.

[6] T. Chantem, Y. Xiang, X. Hu, and R. Dick, "Enhancing multicore reliability through wear compensation in online assignment and scheduling," in *Proc. Conf. on Design, Automation & Test in Europe (DATE)*, 2013, pp. 1373–1378.

[7] D. Gnad, M. Shafique, F. Kriebel, S. Rehman, D. Sun, and J. Henkel, "Hayat: Harnessing Dark Silicon and variability for aging deceleration and balancing," in *Proc. of Design Automation Conf. (DAC)*, 2015, pp. 1–6.

[8] J. Sun, R. Lysecky, K. Shankar, A. Kodi, A. Louri, and J. Roveda, "Workload Assignment Considering NBTI Degradation in Multicore Systems," *Journal Emerg. Technol. Comput. Syst.*, vol. 10, no. 1, pp. 4:1–4:22, Jan. 2014.

[9] L. Huang and Q. Xu, "Characterizing the lifetime reliability of manycore processors with core-level redundancy," in *Proc. Int. Conf. on Computer-Aided Design (ICCAD)*, 2010, pp. 680–685.

[10] T. Kim, X. Huang, H.-B. Chen, V. Sukharev, and S. X.-D. Tan, "Learning-Based Dynamic Reliability Management For Dark Silicon Processor Considering EM Effects," in *Proc. Conf. on Design, Automation & Test in Europe (DATE)*, 2016, pp. 463–468.

[11] JEDEC Solid State Tech. Association, "Failure mechanisms and models for semiconductor devices," *JEDEC Publication JEP122G*, 2010.

[12] A.-M. Rahmani, M.-H. Haghbayan, A. Kanduri, A. Weldezion, P. Liljeberg, J. Plosila, A. Jantsch, and H. Tenhunen, "Dynamic Power Management for Many-Core Platforms in the Dark Silicon Era: A Multi-Objective Control Approach," in *Proc. Int. Symp. on Low Power Electronics and Design (ISLPED)*, 2015, pp. 1–6.

M. Hashem Haghbayan is a PhD student at University of Turku, Finland. He received his MS degree in computer architecture from University of Tehran, Iran. His research interests include high-performance energy-efficient architectures, software-hardware microarchitecture interaction, online/offline testing, formal verification, and reliability.

Antonio Miele is an Assistant Professor at the Politecnico di Milano, Italy. His research interests cover design methodologies for digital systems, in particular focusing dependability issues and runtime resource management in heterogeneous multi-/many-core systems and FPGA-based systems design. He is a member of the IEEE.

Amir M. Rahmani is a University Teacher (Faculty Member) at University of Turku, Finland. His research interests span parallel computer architecture, Internet-of-Things, energy-efficient computing, and runtime resource management. Rahmani received his PhD in Communication and Information Technology from University of Turku, Finland. He is a member of the IEEE.

Pasi Liljeberg is an Adjunct Professor at University of Turku, Finland. His research interests are mainly in the area of embedded systems, multicore processor architectures, Internet of Things, and healthcare technology. Liljeberg received his PhD from University of Turku, Finland. He is a member of the IEEE.

Axel Jantsch is a Professor in the Institute of Computer Technology at TU Wien, Austria. His research interests include dependability, robustness, and self-awareness in SoCs and embedded systems. Jantsch received a PhD in computer science from Vienna University of Technology. He is a member of IEEE.

Cristiana Bolchini received the PhD in Automation and Computing Engineering from Politecnico di Milano, where she is a Professor. Her research interests cover the design and analysis of digital systems with a specific focus on dependability, on which she has authored more than 100 papers. She is a senior member of the IEEE.

Hannu Tenhunen is a Professor in Royal Institute of Technology (KTH), Sweden. His research interests include Network-on-Chip design and embedded systems. He received his PhD degree from Cornell University, Ithaca, NY, USA. He is a member of IEEE.