Analysis of Static and Dynamic Energy Consumption in NUCA Caches: Initial Results

Alessandro Bardine Pierfrancesco Foglia Giacomo Gabrielli Cosimo Antonio Prete
Dip. di Ingegneria dell'Informazione: Elettronica, Informatica, Telecomunicazioni
Università di Pisa
Via Diotisalvi 2, 56122 Pisa (Italy)
{alessandro.bardine, foglia, giacomo.gabrielli, prete}@iet.unipi.it
Members of the HiPEAC European Network of Excellence on
High-Performance Embedded Architecture and Compilation

ABSTRACT

NUCA caches are large L2 on-chip cache memories characterized by multi-bank partitioning and designed to hide wire delay effects. They exhibit high hit rates while keeping access latency low. Proposed designs for such caches are Static NUCA, in which data are statically allocated to the cache banks, and Dynamic NUCA, in which data may reside in different banks, and a migration mechanism is introduced to better tolerate wire delay effects. The two architectures permit to achieve different performances by acting on architectural parameters and data management policies, at the cost of different balances between static and dynamic power consumption and energy dissipation. In this work, we propose preliminary results of the characterization of such balances, by presenting an evaluation of performance and energy consumption of conventional UCAs, and Static and Dynamic NUCA caches. All the considered caches architectures are equal sized and they are supposed to be used in an aggressive high frequency system running some applications from the SPEC CPU2000 and the NAS Parallel Benchmarks suites. The experimental results obtained indicate that, although the migration of data contributes to increase the dynamic energy consumption in Dynamic NUCA caches, the higher IPC achieved permits to save static energy, which dominates the power/energy balance in all the considered architectures. As a consequence, such results would designate NUCA caches as the most performing and energy saving architectures. Besides, according to the obtained results, future power improvements for NUCA caches should concentrate on static energy, while, for the dynamic energy, the on-chip network is the most critical element. Migration of data is acceptable, since it has a positive impact on performance, and the increased dynamic energy is overwhelmed by the static energy savings resulting from the shorter execution time. In order to give a general validity to such statements, we need to explore more design space points for each architecture (by varying the running clock rate and other design parameters) and to evaluate them considering a larger set of benchmarks.

MEDEA '07, September 16, 2007 Brasov, Romania
978-1-9593-807-7/07/09... $5.00

Categories and Subject Descriptors

C.4 [**Performance of Systems**]: measurement techniques, modeling techniques, performance attributes.

General Terms

Measurement, Performance, Design, Experimentation.

Keywords

Cache memories, NUCA, wire-delay, energy consumption, leakage.

1. INTRODUCTION

Technology trends are leading to the use of large, on-chip, level-2 (L2) and level-3 (L3) cache memories. For high frequency systems, the latencies of such caches are dominated by wire delay [1]. In order to reduce the effects of the consequent high access latencies, NUCA Caches (Non-Uniform Cache Architectures) have been proposed [2, 3].

In a NUCA architecture, the cache is partitioned into many independent banks usually interconnected by a switched network (Figure 1); in this model the access latency is proportional to the physical distance of the banks from the cache controller. The mapping between cache lines and banks can be either static or dynamic (namely S-NUCA and D-NUCA); in the former, each line can exclusively reside in a single predetermined bank, while in the latter a line can dynamically migrate from one bank to another. With the last approach, the most frequently accessed data are likely to be located in the closest banks to the cache controller. Both S-NUCA and D-NUCA caches have proven to outperform traditional UCA (Uniform Cache Architecture) caches in large size, wire dominated designs [2, 3]. Various works have been proposed to further optimize their performance, by acting on migration policies [2], on block size and decoupling of tag and data [18], on links and switched network architecture [9, 19]. These studies have been performed both in the single processor and in the CMP domain [21, 22, 23]. All of these works have focused on performance improvements, but none of them have considered power and energy consumption as a main design issue. Nevertheless, a big problem in modern microprocessor design is the rise of total power consumption, a big portion of which, for CMOS processes at 70 nm and below, is due to static power dissipated by leakage currents [24].

Big SRAM structures, like the ones employed in a NUCA-based system, are responsible for a large fraction of the total leakage power budget [24]. Besides, NUCA caches utilize switched networks and data movement policies to achieve high IPC, which further increase power consumption. While the balance between static and dynamic power has been analyzed in the context of conventional UCA caches (leading to different architectural and circuit level optimizations [20, 25, 26, 27]), as of our knowledge, no one has characterized such balance in the context of NUCA architectures.

In this work, we compare achieved performances, level 2 cache's power and energy consumption for a system adopting a conventional level two (L2) UCA cache, with the ones achieved when adopting an L2 S-NUCA or D-NUCA architecture. All the considered caches architectures are equal sized and they are supposed to be used in an aggressive high clock system running some applications from the SPEC CPU2000 and the NAS Parallel Benchmarks suites. For each architecture, we built an energy model including both static and dynamic contributions, highlighting the amount of the different contributions to the total power budget.

Our results for the considered designs and benchmarks indicate that NUCA caches are candidate to be the most performing and energy saving systems. Besides, the static components dominate energy dissipation in all the considered designs. As a consequence, when considering energy consumption, the reduction of static power consumption deserves special attention in the design of NUCA caches as well as for conventional UCA caches. Finally, for the dynamic components, the switched network dissipation is the most critical element in the considered NUCA designs.

We plan to verify the general validity of such results by exploring more design space points for each architecture (by varying the running clock rate and the other design parameters) and evaluating them considering a larger set of benchmarks.

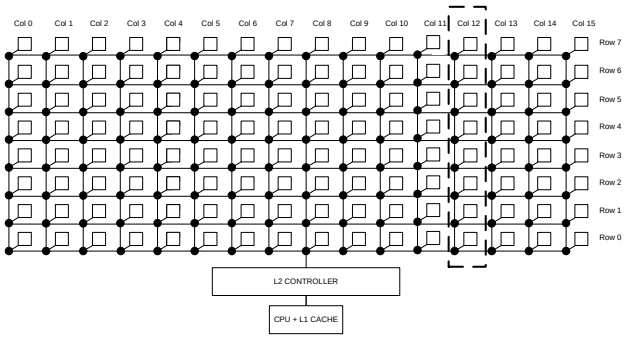

Figure 1: The generic NUCA reference architecture used in our work. The whole space is partitioned into banks (the squares) connected by links and switches (the black circles). In the S-NUCA, each memory address is statically mapped to a single bank; in the D-NUCA, each memory address can be mapped to all the banks of a column (the dashed contours highlight one of such columns). The number of rows and columns may vary, according to the selected configuration.

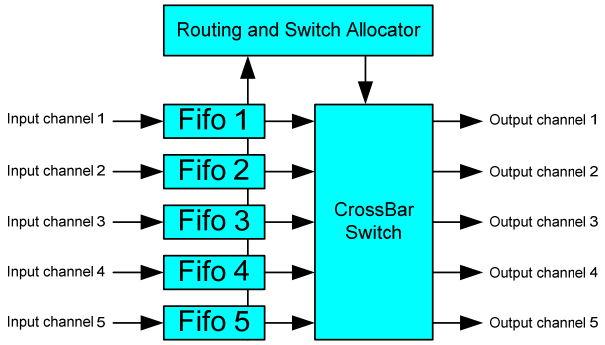

Figure 2: Model of the network switch employed in our NUCA caches. Messages are broken down into multiple flits; the routing of flits is based on a three-stage pipeline composed by a FIFO queue, a routing and switch allocator, and a crossbar switch.

2. EVALUATION OF ENERGY CONSUMPTION

To estimate the energy consumption of the different L2 cache organizations (UCA, S-NUCA and D-NUCA), we developed an energy model that takes into account both static and dynamic contributions. In the following, we describe the reference NUCA architecture on which the model is based (par. 2.1), the energy model (par. 2.2) and the methodology we utilized to get the model parameters (par. 2.3).

2.1 NUCA architecture

Figure 1 shows our generic NUCA reference architecture. The whole memory space is partitioned into banks (the squared boxes in the figure) that are connected to the controller through switches (the black circles in the figure) and links. In the S-NUCA, each memory address is statically mapped to a single bank and searches and other cache references are performed by accessing such bank directly, traversing only those switches and links that connect the controller to the bank. In the D-NUCA architecture, each memory address can be mapped to the group of banks that belong to the same column. When a data search must be performed, the controller first determines the column that can contain the data, then it broadcasts the request to all the banks belonging to such column. To reduce latencies, the "promotion/demotion" mechanism is adopted: if a hit happens in a row that is other than the first, the data line is promoted by swapping it with the line that holds the same column position in the next row closer to the controller.

Cache banks are connected via a packet switching network composed of switches and links. Each message is broken down into multiple flits. The network switch architecture is given in Figure 2. It is built around a three stage pipeline [9, 19] implementing a worm-hole routing algorithm: the first flit of a message arriving from one of the input channels is first buffered in the FIFO queue, then the Routing and Switch Allocator unit calculates its output channel and assigns the crossbar resources, and finally the flit is sent through the crossbar to the output channel. All the subsequent flits of the same message are sent through the same path defined by the first flit. Each link is bidirectional (i.e. it is made up by separate wires for each direction).

From these models, we can derive the main contributors to energy consumption of such memory hierarchies. For the dynamic energy, they are bank accesses, and flits' transmission throughout links and switches. For the static energy, they are banks' and switches' leakage currents (the switched network and its related elements are absent in the UCA model).

In the following, we derive an energy model that describes how such components contribute to the overall energy consumption.

2.2 Energy model

The energy model takes into account both the dynamic and static energy dissipated by the level 2 cache, and the extra dynamic energy consumed when accessing the lower levels of the memory hierarchy on cache misses. This last feature is essential, from an energy consumption perspective, when comparing cache organizations that may exhibit different miss-rates.

If we define *execution time* as the time in seconds needed to execute an application, we evaluate the total energy dissipated during the run of such application according to the formula:

$$E_{total} = E_{dynamic} + E_{static} + E_{off\text{-}cache} \quad \text{(I)}$$

$E_{dynamic}$ is the dynamic energy dissipated by the SRAM banks of the cache and by the network elements (if present), calculated as follows:

$$\begin{aligned} E_{dynamic} = {} & n.\ of\ bank\ accesses \times E_{bank\ access} + \\ & + n.\ of\ flit\ transmissions \times E_{flit\ transmission} + \\ & + n.\ of\ flit\ traversals \times E_{flit\ traversal} \end{aligned} \quad \text{(II)}$$

$E_{bank\ access}$ is the dynamic energy per bank access, $E_{flit\ transmission}$ is the energy required to transmit a flit on a network link, and $E_{flit\ traversal}$ is the energy required to route a flit through a network switch. *N. of bank accesses* is the total number of accesses to the cache banks during execution time, *n. of flit transmissions* is the sum (for all the links) of the number of times each link is traversed by a flit, while *n. of flit traversals* is the sum (for all the switches) of the number of flits routed by each switch during the execution time.

E_{static} is the static energy dissipated by cache banks and network switches due to leakage currents and is calculated as follows:

$$\begin{aligned} E_{static} = {} & (n.\ of\ banks \times P_{bank\ static} + \\ & + n.\ of\ switches \times P_{switch\ static}) \times execution\ time \end{aligned} \quad \text{(III)}$$

where $P_{bank\ static}$ and $P_{switch\ static}$ are the static power consumption that affect cache banks and network switches; *n. of banks* and *n. of switches* respectively represent the total number of cache banks and switches that make up the cache.

$E_{off\text{-}cache}$ represents the dynamic energy dissipated during off-cache accesses due to cache misses:

$$E_{off\text{-}cache} = n.\ of\ off\text{-}cache\ accesses \times E_{off\text{-}cache\ access} \quad \text{(IV)}$$

where $E_{off\text{-}cache}$ access is the energy per off-cache access and *n. of off-cache accesses* is the total number of accesses to the DRAM performed to solve cache misses.

2.3 Energy model parameters

As already observed, the basic elements of the L2 cache architectures considered in our study are memory banks, network switches and network links.

We derived the energy parameters for the memory banks from the CACTI 4.2 tool [7]: we modeled each bank as a whole to obtain its energy consumption per access ($E_{bank\ access}$), its static power consumption ($P_{bank\ static}$) and its physical dimensions. According to the CACTI model, both the parameters include the consumption of the data and tag arrays and of the local control logic of the bank.

For network links, we calculated the energy required for each transmission ($E_{flit\ transmission}$):

$$E_{flit\ transmission} = \alpha \times n.\ of\ wires\ per\ link \times E_{wire} \quad \text{(V)}$$

where α is the switching activity factor, *n. of wires per link* is the link width in bits, and E_{wire} is the energy spent to load the capacitance of a single wire. E_{wire} was calculated adopting a simple RC model [10]; we referred to the Berkeley Predictive Technology Model [11] to derive wire resistance and capacitance per unit length. The length of each link, and thus its total resistance and capacitance, was calculated according to the physical dimensions of cache banks derived from CACTI.

Table 1. Characterization of the benchmark applications included in the study

Benchmark	Suite	FFWD	RUN
bzip2	SPECINT2000	744M	1.0B
gcc	SPECINT2000	2.367B	300M
mcf	SPECINT2000	5.0B	200M
parser	SPECINT2000	3.709B	200M
perlbmk	SPECINT2000	5.0B	200M
twolf	SPECINT2000	511M	200M
applu	SPECFP2000	267M	650M
art	SPECFP2000	2.2B	200M
galgel	SPECFP2000	4.0B	200M
mesa	SPECFP2000	570M	200M
mgrid	SPECFP2000	550M	1.06B
bt	NAS	800M	650M
cg	NAS	600M	200M
sp	NAS	2.5B	200M

For the network switches, the energy parameters were derived from [9], where an energy evaluation for a 3-stage switch architecture like the one utilized in our work is performed with a gate level analysis. From that work, we obtained the following data: energy consumption per flit traversal ($E_{flit\ traversal}$) and switch static power ($P_{switch\ static}$).

For off-cache energy, we assumed that the L2 cache is backed by off-chip DRAM memory; the term $E_{off\text{-}cache}$ corresponds to the energy dissipated by the DRAM memory during off-chip accesses. We derived the energy dissipated on each main memory access ($E_{off\text{-}cache\ \ access}$) from DRAM module datasheets [12], assuming a modern DDR2 system. With the same approach taken by previous works [4][13], we focused our evaluation on the energy dissipated during active cycles (read/write cycles), isolating it from the background energy that is associated to each DRAM power state.

Since leakage power greatly depends on temperature, we scaled each static power term employed in our model assuming 3 selected operating values for the L2 cache: 100°C, 80°C and 60°C. We performed temperature scaling according to the model implemented in the HotLeakage tool [14].

3. METHODOLOGY

The methodology used to perform our evaluation is synthesized in Figure 3.

The selected workload is fed to *sim-alpha* [15], an execution-driven simulator which was extended to support S-NUCA and D-NUCA caches. The resulting execution statistics and the energy parameters are taken as inputs by our energy model to derive total energy consumption.

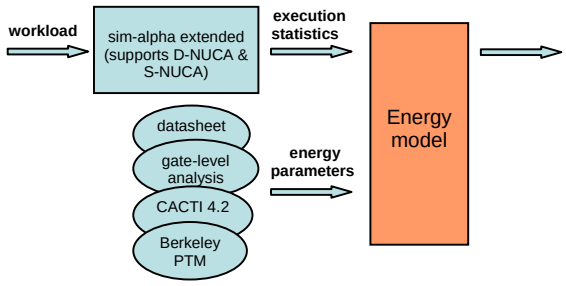

Figure 3. Methodology workflow adopted in the evaluation: execution statistics for the chosen benchmarks are collected by sim-alpha and then combined with the energy parameters in order to get the energy model and finally the energy consumption.

To perform our evaluation, we used the same applications from SPEC CPU2000 [5] and NAS Parallel Benchmarks [6] suites that have been used for the original evaluation of the NUCA caches [2]. These benchmarks were selected for their high L1 miss-rates, so as to achieve a noticeable activity for the big L2 caches considered in this study. For each benchmark we simulated the portion which represents the core repetitive phase of the application. This latter was determined empirically plotting the miss-rate over one execution of each benchmark: the smallest portion of instructions that captured the repetitive behavior of the application was selected. This procedure has already been applied in [2]. Table 1 lists the chosen applications, the number of instructions skipped to reach the phase start (FFWD) and the number of instructions simulated (RUN).

To estimate the switching activity on wires (α) we performed an analysis of the traffic in order to determine the correlation existing between subsequent flits sent on the same link.

4. EXPERIMENTAL CONFIGURATION

We performed our evaluation with a baseline configuration targeted at 70 nm technology. We considered a single processor system with microarchitectural parameters matching those of the Alpha 21264 processor [16], including issue width, fetch bandwidth, and clustering. We selected a clock frequency equal to 8FO4 [17], that roughly corresponds, for 70 nm technology, to a 5 GHz operating frequency.

We assumed that the processor is backed by a 64 KB 2-way set associative L1 I-cache with single cycle access latency and by a 64 KB 2-way set associative L1 D-cache with 3 cycles latency. The cache line size is fixed at 64 bytes.

L2 caches were assumed to have 8 Mbytes capacity with line size fixed at 64 bytes. For each typology (UCA, S-NUCA, D-NUCA) the best performing configuration was selected. For UCA and S-NUCA caches we explored the design space by varying the number of banks (and consequently their size) and bank associativity. To obtain the best performing configuration we also apply the sub-banking technique [7] to design cache banks.

For S-NUCA cache, the mapping between blocks and banks is determined by the lowest order bits of the index field. For D-NUCA cache, we assume that broadcast search and 1 bank/1 hit promotion policies [2] are employed, as described in section 2. This choice maximizes the network traffic and the associated dynamic energy consumption. The links of the NUCA switched networks are assumed to be bidirectional, with 128 bit width for each direction. The network employ a worm-hole routing algorithm. The flit size is fixed at 128 bits, so the transmission of a cache block requires 4 data flits and 1 command/address flit. The switching activity has been calculated to be 0.5. Buffer contention and the decomposition of messages into flits are modeled by the simulator.

In all cache experiments, we assumed that the off-chip memory controller resides near the L2 memory controller. Thus, writebacks need to be pulled out of the cache, and demand misses, when the pertinent line arrives, are injected into the cache by the L2 controller, with all contention modeled as necessary.

Bank latencies are derived from the CACTI tool; the link latency is calculated according to the RC model; the switch traversal latency is obtained with the gate level analysis. The DRAM access latency is set to 300 clock cycles; this value is consistent with the average access time of modern DRAM subsystems.

The configuration parameters are reported in Table 2.

In Table 3 are reported the energy parameters related to the L2 cache configurations described above. These parameters were fed to our analytical energy model to derive the total energy consumption.

Table 2. Configuration parameters: the number of banks, sub-banks, rows and columns and the degree of associativity have been chosen in order to get the maximum performance for each cache architecture

	UCA	S-NUCA	D-NUCA
Size	8 MB	8 MB	8 MB
Line size	64 B	64 B	64 B
N. of banks	1	32	128
N. of sub-banks	2	1	1
N. of bank rows	1	8	16
N. of bank columns	1	4	8
Bank size	8 MB	256 KB	64 KB
Bank associativity	4-way s. a.	4-way s. a.	direct mapped
Bank latency (cycles)	18	5	3
Link latency (cycles)	-	2	1
Link width (bits)	-	2x128	2x128

Table 3. Energy and power parameters for the various cache designs

D-NUCA	$E_{bank\ access}$	41.5 pJ
	$P_{bank\ static}$	235.6 mW (100°C), 137.4 mW (80°C), 70.7 mW (60°C)
	$P_{switch\ static}$	23.1 mW (100°C), 13.5 mW (80°C), 6.93 mW (60°C)
	$E_{flit\ transmission}$	1.8 pJ (vertical link), 6.0 pJ (horizontal link)
	$E_{flit\ traversal}$	135 pJ
	$E_{off\text{-}cache\ access}$	12200 pJ
S-NUCA	$E_{bank\ access}$	287.8 pJ
	$P_{bank\ static}$	1007.0 mW (100°C), 587.1 mW (80°C), 302.1 mW (60°C)
	$P_{switch\ static}$	23.1 mW (100°C), 13.5 mW (80°C), 6.93 mW (60°C)
	$E_{flit\ transmission}$	5.68 pJ (vertical link), 6.87 (horizontal link)
	$E_{flit\ traversal}$	135 pJ
	$E_{off\text{-}cache\ access}$	12200 pJ
UCA	$E_{bank\ access}$	1928.5 pJ
	$P_{bank\ static}$	32205.9 mW (100°C), 18776.0 mW (80°C), 9661.8 mW (60°C)
	$E_{off\text{-}cache\ access}$	12200 pJ

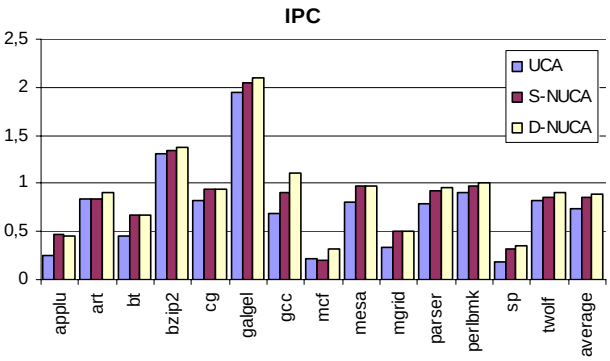

Figure 4. Comparison of the IPC achieved by the three considered cache architectures: D-NUCA outperforms S-NUCA by 4.98% and UCA by 20.77% (average values).

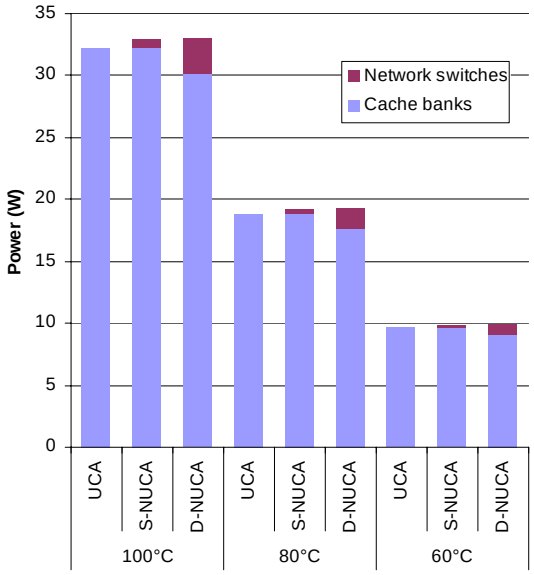

Figure 5. Breakdown of the static power consumption of the simulated L2 caches at various temperature values. D-NUCA is always the most power consuming architecture because of the increased number of network switches it uses compared to S-NUCA.

5. RESULTS

a) Performance

The performance comparison between L2 UCA, S-NUCA and D-NUCA caches is shown in Figure 4. D-NUCA caches outperform UCA caches in all the considered benchmarks, while only for the *applu* benchmark the S-NUCA outperforms D-NUCA. Taking the average IPC across the entire workload as a reference metric, D-NUCA outperforms UCA and S-NUCA by 20.77% and 4.98% respectively.

b) Static components and overall energy consumption

Figure 5 reports the static power consumption of the L2 caches (in Watts) for each configuration. We highlight the components due to cache banks and network switches. The amount of static power due to cache banks varies among the three architectures since the individual bank configurations are different (Table 3). At each temperature, D-NUCA caches exhibit the largest dissipation. This is caused by the static power dissipated by network switches (128 switches are employed in a D-NUCA against only 32 in a

S-NUCA). Considering the total power consumption, the UCA architecture is the most power saving.

The average energy consumed by the three architectures at different operating temperature is given in Figure 6. The energy values were calculated according to the formulas described in paragraph 2 (the static power values were multiplied by the execution time of the workload to achieve the final energy value). We highlight two different contributions: the dynamic contribution due to bank accesses, switched network activity and off-chip accesses, and the static contribution, dissipated by cache banks and network switches.

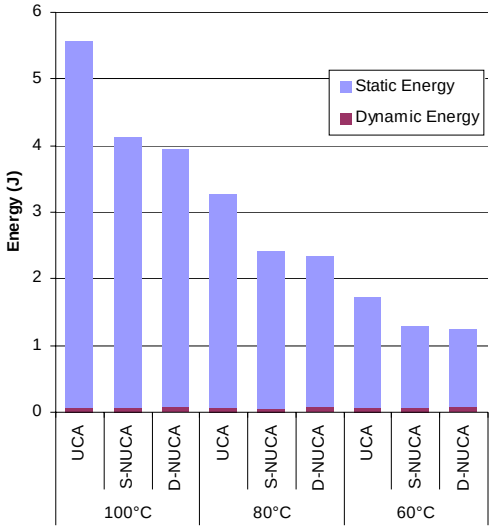

Figure 6. Average energy consumption comparison of the simulated L2 caches at various temperature values. D-NUCA is always the most energy saving architecture. As previously found for UCA, the dynamic and static breakdown shows that also S-NUCA and D-NUCA are dominated by the static components.

The energy profile of the UCA architecture agrees with previous analyses [20]: in such caches the static component dominates the dynamic one. Our analysis extends this result also to NUCA caches: for all the considered architectures, energy consumption is dominated by the static component. The dynamic contribution is almost negligible: the percentage of dynamic energy with respect to the total energy value is 1.63% for UCA, 2.24% for S-NUCA and 3.73% for D-NUCA at an operating temperature of 80°C. Such temperature is representative of the typical working condition of the processor, while the other two may be considered upper and lower limits for such conditions. In particular, the 60°C is the temperature that a processor of the present generation achieves in no load conditions [28, 29].

D-NUCA is the most energy saving architecture at all the considered temperature values. In particular, even if such architecture experiences the highest number of bank accesses (Figure 8a) and flit transmissions (Figure 8b) due to the data search and migration activities, the extra dynamic energy dissipated when performing such activities is overwhelmed by the static energy savings deriving from the shorter execution time due to the higher IPC.

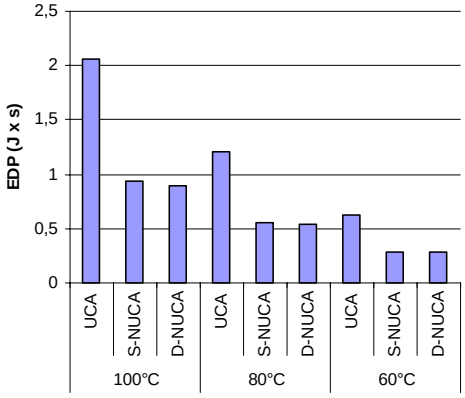

Figure 7. Energy-delay product (EDP) for the three cache architectures at each temperature value (lower is better). NUCA caches always exhibit the lowest EDP.

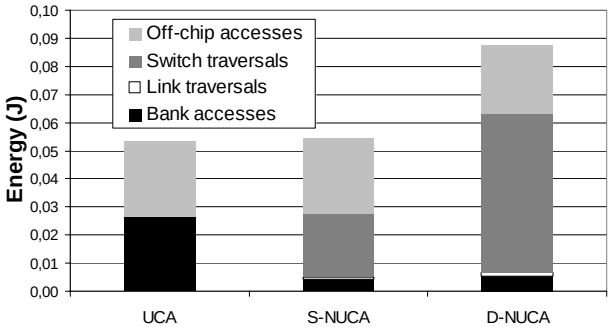

Figure 8. Average dynamic energy consumption for the simulated L2 caches. D-NUCA exhibits the highest consumption. As shown by the components' breakdown, this is mainly due to the increase switch traversals' energy that is consequence of the increase in the switched network traffic (compare Figure 9) while the increase in bank accesses has only marginal effects on the dynamic energy consumption.

While the energy breakdown is useful to compare the different contributions to total energy consumption, the energy depends on performance and is not the only metric to take into account when comparing the goodness of different solutions; as proposed in previous works [29, 30], we also report the energy-delay product (EDP) for the three cache architectures (Fig. 7). This metric allows to evaluate the energy-performance trade-off for the alternative cases. The considered NUCA architectures exhibit lower EDP values with respect to UCA architectures.

c) Dynamic components

Figure 8 shows the breakdown of the average dynamic energy contribution in its various components: bank accesses, link traversals, switch traversals, and off-chip accesses. Figure 8

shows averaged statistics related to bank accesses, flit transmissions and DRAM accesses.

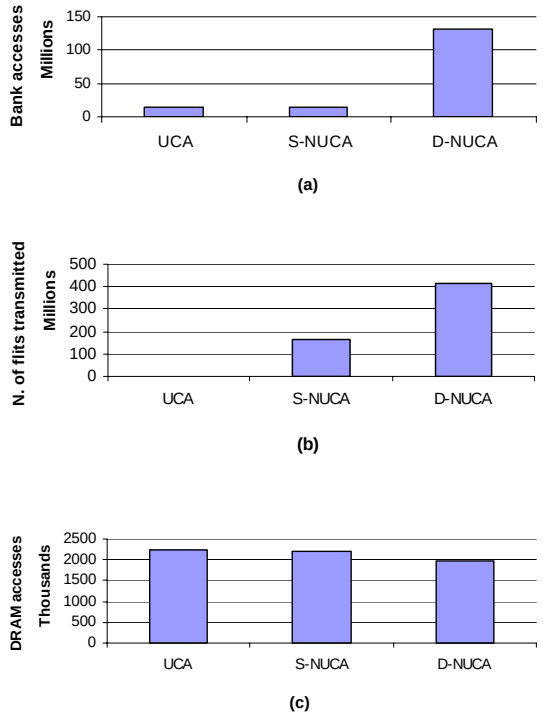

Figure 9. Average number of bank accesses (a), average number of flits transmitted (b), and average number of DRAM accesses (c). The average number of bank accesses (a) is almost identical in UCA and S-NUCA while it grows in D-NUCA because of promotions and migrations; similarly, the average number of flits transmitted over the network (b) grows up by 2.5 times when moving from S-NUCA to D-NUCA (there is no network in UCA). The average number of DRAM accesses (c) decreases when moving from UCA to S-NUCA and to D-NUCA because of the increased hit rate of these latter cache designs.

The D-NUCA architecture exhibits the highest dynamic energy dissipation (1.63 times higher than UCA and 1.60 times than D-NUCA), while UCA and S-NUCA energy consumption are comparable. This is mainly due to the switched network activity. As a consequence of the search and migration mechanisms, the number of flit transmissions (Figure 9b) is maximum for the D-NUCA systems (2.5 times higher than in S-NUCA). This traffic generates the highest component of the dynamic energy consumption. D-NUCA caches exhibit also the highest number of bank accesses (Figure 9a), but such component has little impact on energy, due to the small size of the banks that constitute the cache (Table 2 and 3). The off-chip component of energy is higher in the UCA cache. This is a consequence of the higher miss-rate exhibited by such a cache, which generates the highest number of DRAM accesses (Figure 9c).

6. CONCLUSIONS AND FUTURE WORKS

In this work, we have presented the preliminary results of a comparison of performance and energy consumption of conventional UCA caches and Static and Dynamic NUCA caches. The results obtained for the considered designs and benchmarks, would candidate NUCA caches to be the best performing and the most energy saving architectures.

Besides, the obtained results suggest that, in spite of the highest number of bank accesses and the highest network traffic generated in the D-NUCA caches by data search and migration, the total energy budget is dominated by the static component. It turn out that, as for conventional L2 UCA caches, future power improvements should concentrate on the leakage component. For the dynamic component, the switched network appears to be the most critical element. In any case, migration policies are acceptable as: i) they contribute to achieve the higher IPC, ii) although they introduce extra dynamic energy consumption, in L2 caches, which are dominated by static energy, they contribute (thanks to the increased performance) to lead to the least energy consuming memory system.

In order to give a more general validity to such conclusions, future works will be focused on the exploration of more design space points for each architecture (by varying the running clock rate and the other design parameters such as their number of banks, their size, their associativity, etc.) and to evaluate them considering a complete set of benchmarks suites.

7. ACKNOWLEDGMENTS

This work has been partially supported by the SARC project funded by the European Union under the contract no. 27648.

8. REFERENCES

[1] D. Matzke, “Will Physical Scalability Sabotage Performance Gains?”, *IEEE Computer*, Volume 30, Issue 9, pp. 37-39, Sept. 1997.

[2] C. Kim, D. Burger, and S. W. Keckler, “An Adaptive, Non-Uniform Cache Structure for Wire-Delay Dominated On-Chip Caches”, *Proceedings of the 10th International Conference on Architectural Support for Programming Languages and Operating Systems*, pp. 211-222, San Jose, CA, USA, October 2002.

[3] C. Kim, D. Burger, and S. W. Keckler, “Nonuniform Cache Architectures for Wire-Delay Dominated On-Chip Caches”, *IEEE Micro*, Volume 23, Issue 6, pp. 99-107, Nov.-Dec. 2003.

[4] N. S. Kim, D. Blaauw, and T. Mudge, “Quantitative Analysis and Optimization Techniques for On-Chip Cache Leakage Power”, *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, Volume 13, Issue 10, pp. 1147-1156, Oct. 2005.

[5] *Standard Performance Evaluation Corporation.* [Online]. Available: http://www.spec.org/cpu2000/.

[6] *NAS Parallel Benchmarks.* [Online]. Available: http://www.nas.nasa.gov/Resources/Software/npb.html.

[7] D. Tarjan, S. Shyamkumar, and N. P. Jouppi, *“CACTI 4.0”*, HP Technical Report, HPL-2006-86, June 2006.

[8] H. Wang, L. Peh, and S. Malik, “A Power Model for Routers: Modeling Alpha 21364 and InfiniBand Routers”,

IEEE Micro, Vol. 23, No. 1, pp. 26-35, January/February 2003.

[9] N. Muralimanohar, and R. Balasubramonian, "Interconnect Design Considerations for Large NUCA Caches", *Proc. of the 34th International Symposium on Computer Architecture* (ISCA), pp. 369-380, San Diego, CA, June 2007.

[10] N. Weste, and D. Harris, *CMOS VLSI Design: A Circuits and Systems Perspective (3rd edition)*, Addison Wesley, New York, 2004.

[11] *Berkeley Predictive Technology Model* [Online]. Available: http://www.eas.asu.edu/~ptm/.

[12] *Micron 1 GB DDR2 SDRAM Module Datasheet*. [Online]. Available: http://www.micron.com.

[13] V. Delaluz, et al., "Compiler-Directed Array Interleaving for Reducing Energy in Multi-Bank Memories", *Proceedings of Asia and South Pacific Design Automation Conference* 2002, pp. 288-293, Bangalore, India, January 2002.

[14] Y. Zhang, et al., *"HotLeakage: A Temperature-Aware Model of Subthreshold and Gate Leakage for Architects"*, University of Virginia, Department of Computer Science Technical Report, CS-2003-05, March 2003.

[15] R. Desikan, et al., *"Sim-alpha: a Validated, Execution-Driven Alpha 21264 Simulator"*, University of Texas at Austin, Department of Computer Sciences Technical Report, TR-01-23, 2001.

[16] R. E. Kessler, E. J. McLellan, and D. A. Webb, "The Alpha 21264 Microprocessor Architecture", *Proceedings of the International Conference on Computer Design*, p. 90, Austin, TX, USA, October 1998.

[17] M. S. Hrishikesh, et al., "The Optimal Logic Depth Per Pipeline Stage is 6 to 8 FO4 Inverter Delays", *Proceedings of the 29th International Symposium on Computer Architecture*, pp. 14-24, Anchorage, AK, USA, May 2002.

[18] Z. Chisti, M. D. Powell, and T. N. Vijaykumar, "Distance Associativity for High-Performance Energy-Efficient Non-Uniform Cache Architectures", *Proceedings of the 30th International Symposium on Microarchitecture*, pp. 55-66, San Diego, CA, USA, Dec. 2003.

[19] J. Yuho, K. Jung, and Y. Hwan, "A Domain-Specific On-Chip Network Design for Large Scale Cache Systems", *Proceedings of High Performance Computer Architecture*, pp. 318-327, Phoenix, AZ, USA, Feb. 2007.

[20] H. Hanson, M. S. Hrishikesh, V. Agarwal, S. W. Keckler, and D. Burger, "Static Energy Reduction Techniques for Microprocessor caches", *IEEE Transactions on VLSI*, vol. 11, n. 3, pp. 303-313, June 2003.

[21] J. Hu, C. Kim, H. Shafi, L. Zhang, D. Burger, and S. W. Keckler, "A Nuca Substrate for Flexible CMP Cache Sharing", *IEEE Transactions on Parallel and Distributed Systems*, vol. 18, n. 8, pp 1028-1040, August 2007.

[22] Z. Chishti, M. D. Powell, and T. N. Vijaykumar, "Optimizing Replication, Communication, and Capacity Allocation in CMPs", *Proceedings of the 32th International Symposium on Computer Architecture*, pp. 357-368, Madison, WI, USA, June 2005.

[23] B. M. Beckmann, and D. A. Wood, "Managing Wire Delay in Large Chip-Multiprocessor Caches", *Proceedings of the 37th International Symposium on Microarchitecture*, pp. 319-330, Portland, OR, USA, Dec. 2004.

[24] N. S. Kim, *et al.*, "Leakage Current: Moore's Law Meets Static Power", *IEEE Computer*, vol. 36, n. 12, pp. 68-75, Dec. 2003.

[25] N. S. Kim, *et al.*, "Drowsy Instruction Caches: Leakage Power Reduction Using Dynamic Voltage Scaling and Cache Sub-bank Prediction", *Proceedings of the 35th Annual International Symposium on Microarchitecture*, pp. 219-230, Instanbul, Turkey, Nov. 2002.

[26] V. Venkatachalam, and M. Frank, "Power Reduction Techniques for Microprocessor Systems", *ACM Computing Survey*, vol. 37, n. 3, pp. 195-237, Sept. 2005.

[27] Y. Meng, T. Sherwood, and R. Kastner, "Exploring the Limits of Leakage Power Reduction in Caches", *ACM Transactions on Architecture and Code Optimizations*, vol. 2, n. 3, pp 221-246, Sept. 2005.

[28] Intel Corporation, Intel® Pentium® 4 Processor Extreme Edition 3.73 GHz processor specification. *Datasheet available at www.intel.com.*

[29] R. Gonzalez, and M. Horowitz, "Energy Dissipation in General Purpose Microprocessors", *IEEE Journal of Solid-State Circuits*, vol. 31, no. 9, pp. 1277-1284, September 1996.

[30] H. Hanson, M.S. Hrishikesh, V. Agarwal, S. W. Keckler, and D. Burger, "Static Energy Reduction Techniques for Microprocessor Caches", *Proceedings of the 2001 Int. Conference on Computer Design*, pp.276-283 , Austin, TX, USA, September 2001.