Big versus Little: Who will trip?

Reena Panda
University of Texas at Austin
reena.panda@utexas.edu

Christopher Donald Erb
University of Texas at Austin
cde593@utexas.edu

Lizy Kurian John
University of Texas at Austin
ljohn@ece.utexas.edu

***Abstract*—Since the marginal cost of operating powerful monolithic single core systems has become prohibitive, horizontal scaling has become the de-facto method for expanding computational power and maintaining acceptable levels of energy efficiency. While horizontal scaling is now the accepted means, there is still a debate as to whether this should be done with "big" or "little" architectures. While this subject has typically been approached from the perspective of performance or power, we choose to analyze it in the light of reliability. In recent years reliability has joined performance and power as a first-order design constraint in microprocessor design. The sensitivity of microprocessors to voltage fluctuations is a major concern in designing efficient low-power, reliable micro-architectures. Voltage fluctuations beyond a certain threshold can cause timing errors and operational failures in processors, risking the reliability of systems. While this has traditionally been studied in the context of few-core systems, compounding effects may be experienced by larger parallel and distributed systems which have become the mainstream in desktop/server class computing. In this paper, we perform a detailed evaluation of the characteristics of voltage noise in large many-core systems, comparing the differences in future many-core out-of-order (OOO) and inorder configurations. We find that single out-of-order cores experience larger voltage variations when compared to inorder cores, but also have a clear advantage in terms of performance. Based on our evaluation using parsec benchmarks, we find that for processes that scale with the number of cores, a number of OOO cores may be replaced by a larger number of inorder cores to achieve the same power-efficiency and performance with improved reliability.**

***Keywords*-Reliability; Voltage Noise; Out-of-order cores; In-order cores; Power Efficiency**

I. Introduction

Today microprocessor designs are constrained more by power efficiency than by performance. This has led to a proliferation of design techniques for improved power efficiency, starting from a renewed interest in smaller power-efficient inorder cores, to employing dynamic power management techniques to reduce power consumption. Such power-saving techniques are employed to save power wherever and whenever possible. The decision to pursue power efficiency in either the avenue of small inorder cores or larger OOO cores has re-ignited the big-little debate. A few big cores or many small cores? Many would choose big cores, this consolidates the system and removes complications created when several discrete processors need to coordinate their actions but comes with added internal complexity. As we will show in this paper, this added complexity has its own issues. More recently, performance and power constraints have begun to wear on system components effectively stringing out a trip-line for reliable operation. Aggressive power saving techniques, like clock gating [1] and dynamic voltage/frequency scaling [2], can cause large variations in supply current by throttling workload activity over small periods of time. Due to the parasitic impedance in the power delivery network, these rapid changes in load current cause supply voltage fluctuations (typically referred to as voltage noise) from the nominal value. Such voltage fluctuations are dangerous because if the supply voltage crosses the tolerance limits, the chip is susceptible to malfunction. Hence, reliability is no longer an assumption, but has become a first-order design constraint. In this paper, we assess the big-little debate from a reliability perspective.

A number of studies [3], [4], [5] have characterized the impact of voltage noise in microprocessors but they have primarily focused on uniprocessor systems or few-core chip multi-processor (CMP) systems. Given the increasing relevance of large multi-core systems, we perform a detailed characterization of voltage noise behavior in CMPs, consisting of large number of cores. Furthermore, prior research has studied voltage noise only in performance-oriented OOO cores. With the increased adoption of small, power-efficient inorder cores in systems ranging from mobile devices to servers, it is critical to understand if there is a difference in the nature of voltage noise between the two types of cores. While the big-little debate is not new, it has typically been dealt with from the perspective of either performance or power-efficiency [6], [7]. In this paper we take from the vantage point of reliable operation. The questions we seek to answer from the analysis are:

- How does the voltage noise behavior change as number of cores are scaled in large multi-core systems?
- Are any voltage-noise compounding effects experienced due to interactions among the multiple core and uncore components in larger multi-core systems?
- How do the voltage noise behaviors differ in inorder and out-of-order based multiprocessor systems? Is one better than the other?

This paper presents a comparative study of voltage noise in CMPs consisting of high-performance out-of-order cores

and power-efficient inorder cores. Our results highlight that single OOO cores experience much larger voltage variations when compared to the inorder cores, but offer a clear advantage in terms of performance. We find that as the number of cores are scaled in multiprocessor systems, OOO CMPs experience much higher voltage swings as compared to inorder CMPs and thus, are more susceptible to reliability issues. Our experiments further indicate that iso-power inorder CMP configurations that offer equivalent performance as OOO CMP configurations offer much lower voltage noise and thus, improved reliability characteristics. We compare the performance, voltage noise, and energy-efficiency of CMP organizations with different types of cores. These analyses can provide important insights and prove very valuable in designing low-power, reliable multiprocessor systems in the future. Our evaluation can also enable efficient exploration of resilient architecture designs that allow systems to run with aggressive voltage guardbands [8], [9], [10], [11] and employ recovery circuits to detect/correct operational failures stemming from voltage emergencies.

The paper is organized as follows: In Section 2, we describe our experimental setup and methodology. Section 3 describes our results and analyses in detail. Finally, we conclude the paper in section 4.

II. Simulation Methodology

In this section, we describe our experimental methodology in detail.

A. Simulation Infrastructure

We use a full-system simulator, marssx86 [12] for our experiments. We use a modified version of McPAT [13] for performing power studies. The configuration parameters for the single out-of-order and inorder core are shown in Table I. Multicore OOO configurations use a 3-level cache hierarchy, with the shared L3 cache size being scaled as the number of cores is increased. The inorder core configurations use 2-levels of cache, with the size of L2 scaled with the number of cores.

Table I: Core Configurations

	Out-of-Order Core	In-Order Core
Clock Rate	3.0 GHz	1.6GHz
Fetch Width	4	2
Decode Width	4	2
Inst. Window	128 ROB, 64 LSQ	-
BTB	1024 Entries	1024 Entries
RAS	1024 Entries	1024 Entries
L1 I/D Cache	32 KB each, 4-way, 2 cycles	32 KB each, 4-way, 2 cycles
L2 Cache	256 KB, 8-way, 12 cycles	256 KB, 8-way, 12 cycles
L3 Cache	1MB, shared, 40 cycles	-
Int. ALU and Mult/Div	2 per core, 1 cycle	2 per core, 4 cycles
FP ALU	2 per core, 6 cycles	

B. Integration of McPat and Marssx86

We use an integrated performance-power model infrastructure, called pvSim [14] that integrates a modified version of McPAT with marssx86 simulator to obtain per-cycle power statistics. pvSim uses a modified version of Mcpat that removes McPat's XML interface and builds it as a library which is linked with the Marssx86 simulator as a power hook. Marssx86 simulator is used to simulate the benchmarks and per-cycle statistics are fed from marssx86 simulator to McPat, which then generates the per-cycle power trace (based on 45nm technology). For events that take more than one cycle to complete, like ALU operations, cache events etc, the pvSim model distributes the power evenly across multiple cycles. We model the power consumed by the core, private and shared caches. We do not include power consumption by other components, like the memory controller and interconnects, as previous studies [9] have shown that voltage variations are not very sensitive to load variations in these components.

C. Power and Voltage Modeling

Large variations in the current drawn from the power delivery network (PDN) cause inductive noise in the chip, whose magnitude depends on the characteristics of the PDN. For our experiments, we use a second-order lumped model [15]. The PDN is modeled based on the parameters of the Pentium 4 package and its characteristics are summarized in Table II. The PDN is kept the same as the number of cores are varied, to demonstrate the impact of increase in core count on the magnitude and frequency of voltage variations. With a supply voltage of 1V, the power estimates are convolved with an impulse response of the power supply network to obtain the voltage variations at per-cycle granularity. One of the limitations of the lumped voltage model is that it does not capture local, inter-core voltage variations in a CMP, but instead provides an aggregate view of the voltage variations across the entire chip. A distributed voltage model, using a RL network to model the cores and functional units in the core at a much finer granularity, has thus been proposed in literature [16] to capture inter-core voltage variations. Nevertheless, for this paper, the lumped model is sufficient as our goal is to study voltage noise characteristics at a higher package level.

D. Benchmarks

We use the multi-threaded PARSEC benchmarks [17] for our experiments. We run all of the parsec benchmarks except canneal due to simulation time constraints. Each PARSEC benchmark is run for 100 million instructions

Table II: PDN Parameters Used

Resonant frequency	Peak impedance	Quality factor
100 MHz	2.25mΩ	3

from the region of interest using the simlarge input set. The number of threads of execution equals the number of simulated cores and is affined to a core. We do not show the results for facesim and fluidanimate benchmarks for the inorder10 and OOO3 configurations because these benchmarks can run with an even or power-of-two number of threads respectively.

III. Experimental Results

In this section, we discuss our analysis of voltage noise behavior in big and little cores.

A. Characterization of voltage noise in OOO core configurations

This section presents a detailed characterization of voltage noise in different OOO core configurations.

Figure 1 shows the distribution of samples for different magnitudes of voltage swings for the PARSEC benchmarks on a single OOO core. We can observe that different benchmarks result in different voltage swing behavior in the OOO core, which implies that the benchmarks experience different levels of activity fluctuations. It can however, be seen that the majority of the samples are distributed close to the nominal supply voltage and a very small percentage of all the samples exceed 1% of undershoot. Only bodytrack and vips experience a maximum voltage drop of greater than 2%. Thus, for our experiments, we assume an aggressive voltage margin of 2%, purely for characterization purposes.

Figure 2 shows the maximum voltage swing for each benchmark, as the number of OOO cores are increased from 1 to 8. We can observe that as the number of cores increase, the maximum worst case drop increases as well. The magnitude of maximum voltage swing increases from 2.28% to 8.82% from 1-core to 8-cores. This trend demonstrates interference among the micro-architectural activity across

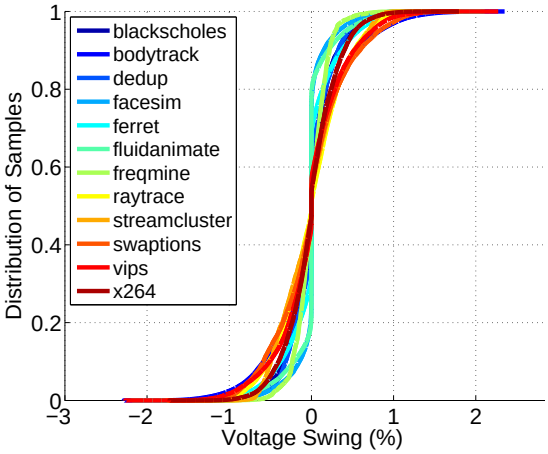

Figure 1: Cumulative distribution of voltage swings on a single OOO core

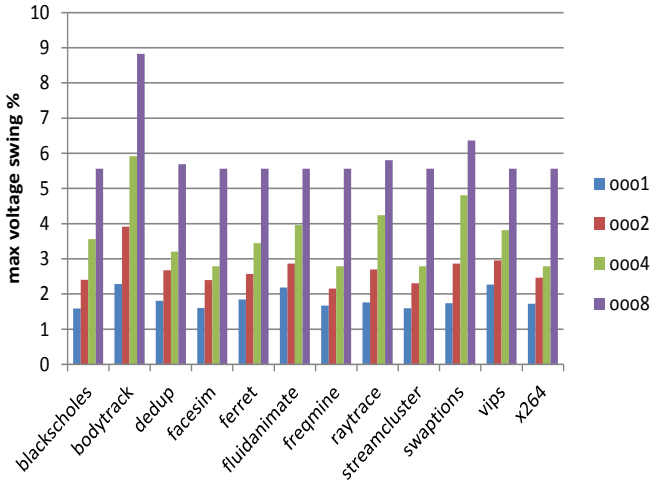

Figure 2: Impact of increase in core count on maximum voltage undershoot in OoO cores

cores that causes larger voltage swings than the single-core counterparts. As compared to a single-core configuration, the bigger core systems have a higher percentage of samples exceeding the assumed voltage margin values. For example, the number of samples exceeding the voltage margins increases by over 200% from a 1-core to a 8-core CMP for bodytrack benchmark.

B. Characterization of Voltage Noise in inorder core configurations

This section presents a characterization of voltage noise on inorder core-based CMP configurations.

Figure 3 shows the distribution of samples of voltage swings for the PARSEC benchmarks in a single inorder core. We can clearly observe that the magnitude of voltage swings experienced by the single inorder core is much lower than a single OOO core. Again, different benchmarks result in different levels of maximum voltage swings in inorder cores. It can also be seen that the majority of samples are distributed close to the nominal supply voltage and none of the samples exceed the 2% of undershoot for a single inorder core.

Figure 4 shows the impact of increasing core counts on the observed voltage swings of inorder CMPs. We can observe that maximum voltage swing increases as the number of cores are increased from 1 to 8, however the magnitude of voltage swings is much lower as compared to OOO CMPs. Also, as the number of cores increase, a higher percentage of samples exhibit higher voltage swings. It can also be observed that many parsec benchmarks experience similar maximum voltage swings but at different periods of their execution. This might be attributed to the nature of the inorder pipeline, where the pipeline stalls if there is a resource conflict or in the event of cache misses and, as a

Table III: TDP Equivalence across different CMP configurations

	OOO	Inorder	TDP
Config-I	1	4	52-55W
Config-II	2	8	94-100W
Config-III	3	10	128-137W

result, all the benchmarks experience periods of execution followed by periods of stalls, leading to similarity in the overall voltage noise behavior.

C. Inorder vs OoO : A Reliability Perspective

The big out-of-order cores and small inorder cores differ in the way they execute the dynamic instruction stream. In this section, we compare the maximum voltage swings experienced by inorder and OOO CMP configurations as the core counts increase. Figure 5 indicates a very interesting trend in the rate of increase of the magnitude of the worst case voltage swing for the two types of cores. We can observe that the magnitude of voltage swings increases in both cases as the core count increases, however the inorder configurations experience much lower swings than OOO configurations even with their 8-core systems. Also, the rate of increase in the magnitude of voltage swings in inorder cores is much slower as compared to OOO cores. These trends have strong implications on the design of future servers composed of large number of inorder cores based on better reliability characteristics.

D. Voltage Noise characteristics in TDP Equivalent systems

This section analyzes voltage noise in inorder and OOO CMPs from the perspective of the thermal design power values. The thermal design power (TDP) indicates the maximum amount of heat generated by the CPU that the cooling system is required to dissipate when running typical

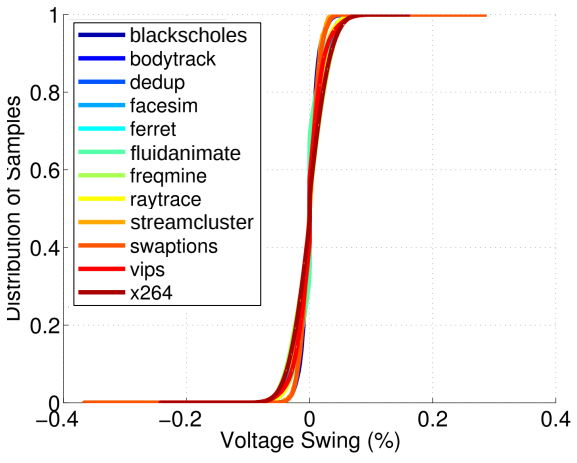

Figure 3: Cumulative distribution of voltage swings on a single inorder core

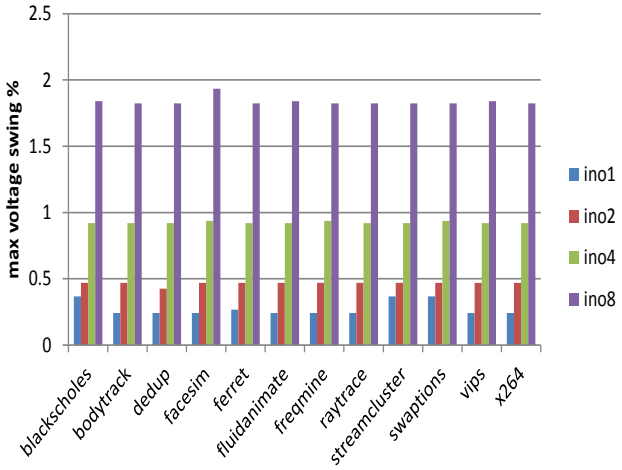

Figure 4: Impact of increase in core count on maximum voltage undershoot in inorder core CMP

"real-world" applications. The PDN of a microprocessor is designed taking into account the designated peak power of the processor. The peak power of a multi-core system varies

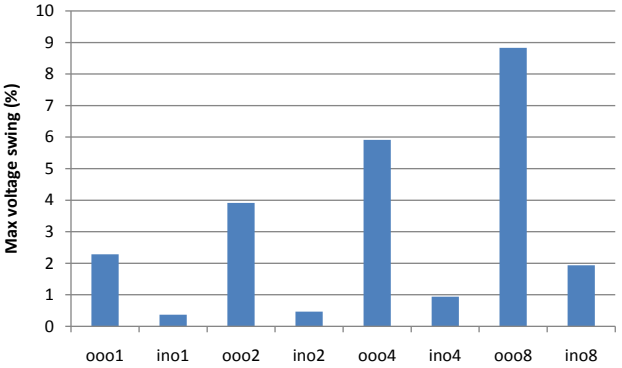

Figure 5: Voltage swing comparison between OOO and inorder cores

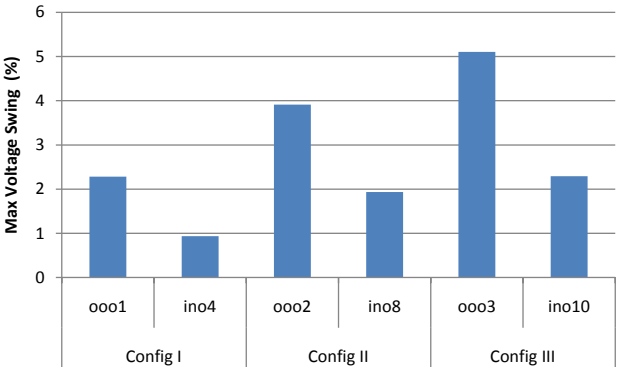

Figure 6: Comparison of maximum voltage swings across TDP equivalent configurations

as the total number of cores vary. Thus, to have a fair comparison of the level of voltage noise across different multi-core configurations comprising of different types of cores, we compare configurations with the same designated peak power as reported by mcpat. The TDP equivalent configurations considered in this section are summarized in Table III. The mapping of OOO to inorder cores is not linear due to different sizes of the last-level caches.

Figure 6 shows the maximum voltage swing in TDP-equivalent OOO and inorder configurations. For TDP equivalent configurations, inorder cores experience much lower maximum voltage swings than OOO cores and can be operated using more aggressive voltage margins without risking reliability. Aggressive voltage margins can translate to (a)reduction in supply voltages, thereby improving power requirements or (b)higher operating frequencies, thereby improving performance further.

1) Performance comparison for TDP equivalent systems: In the past, power/energy-efficiency were traded off for improved performance. But such trade-offs are hardly opted for anymore. When designing today's computer systems anywhere from embedded devices like smart-phones to huge data-centers, performance per watt and energy-efficiency are the metrics that are talked about. In that light, here we compare the performance and voltage noise behavior of different inorder and OOO CMPs for the iso-power (TDP equivalent) configurations. Figure 7 shows the performance equivalence between the two types of cores. We can observe that for many parsec benchmarks, the bigger inorder

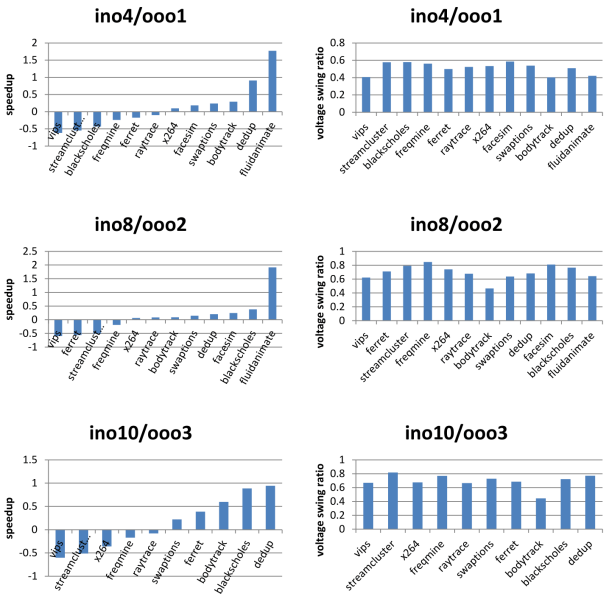

Figure 7: Performance and Voltage Noise Comparison of TDP equivalent CMPs

configurations can achieve comparable or better performance than fewer OOO cores. This is because parsec benchmarks are multi-threaded and can scale in terms of performance as the number of threads are scaled up. For instance, with 4 inorder cores, about 50% of the PARSEC benchmarks yield comparable/better performance when compared to a single OOO core. So, in terms of performance, for some of the PARSEC benchmarks, a variable number of inorder cores can be used in lieu of the more power-hungry OOO cores while achieving the same/better power-efficiency. For the benchmarks that perform well on larger inorder core configurations, it translates to improved energy-efficiency and reliability.

However, for the benchmarks that do not scale as well with the number of cores, fewer high-performance OOO cores perform better as compared to larger number of inorder cores. Thus, running such applications on larger number of inorder cores would result in poor performance and energy efficiency. For those benchmarks which see significant slowdown on larger inorder CMP configurations, the benefits of using inorder cores to match the performance of corresponding OOO cores might get nullified. However, even for such benchmarks, the inorder core configurations result in much better reliability characteristics than OOO configurations. These larger inorder configurations can be run with more aggressive voltage margins, which can translate to better power-efficiency (lower supply voltages) or higher performance (higher operating frequencies). Moreover, the

IV. Conclusion

In this paper, we have presented a detailed characterization of voltage noise effects in large multi-core systems. In the light of renewed interest in smaller inorder processors for designing computer systems, we have also presented a detailed evaluation of how the voltage noise effects differ in OOO and inorder cores. Our results demonstrate that as the number of out-of-order cores increase, the magnitude of the worst-case voltage droop increases, while in the case of inorder cores, the worst-case swings also increase but at a much slower rate. Our evaluations comparing iso-power out-of-order core configurations and inorder core configurations showed that larger numbers of inorder cores have better voltage noise behavior, while having comparable or better performance than fewer-core out-of-order systems on a number of parsec benchmarks. This implies that micro-architectures designed for worst-case voltage noise will require very large voltage guard-bands on out-of-order systems, resulting in wastage of power and reduced peak operating frequency. Our results also show that the frequency of the worst-case swings is much lower for inorder core systems, less than 0.1%, and is not significantly impacted as the number of cores increase, indicating the feasibility of micro-architecture designs that are optimized for typical case behavior. We thus conclude that CMP designs with

inorder cores are more favorable than OOO core designs in terms of reliability, with smaller and less frequent voltage swings. For many parallelizable/scalable parsec benchmarks, the iso-power inorder core configurations yield comparable or better performance to OOO cores, implying improved energy-efficiency as well. There are times when inorder CMPs are outperformed by OOO CMPs because they are limited by the scalability of the program, but this may still be less important when reliable operation is a top priority.

ACKNOWLEDGMENT

This material is based upon work supported by NSF grants 1117895, 1218474, Semiconductor Research Corporation task 2014-HJ-2504. Any opinions, findings, and conclusions or recommendations expressed herein are those of the authors and do not necessarily reflect the views of the NSF or SRC.

REFERENCES

[1] Q. Wu, M. Pedram, and X. Wu, "Clock-gating and its application to low power design of sequential circuits," *Proc. of the IEEE Custom Integrated Circuits Conference*, vol. 47, pp. 415–420, 2000.

[2] M. Weiser, B. Welch, A. Demers, and S. Shenker, "Scheduling for reduced cpu energy," *USENIX SYMP. OPERATING*, pp. 13–23, 1994.

[3] V. J. Reddi, S. Kanev, W. Kim, S. Campanoni, M. D. Smith, G.-Y. Wei, and D. Brooks, "Voltage smoothing: Characterizing and mitigating voltage noise in production processors via software-guided thread scheduling," in *Proceedings of the 2010 43rd Annual IEEE/ACM International Symposium on Microarchitecture*, ser. MICRO '43. Washington, DC, USA: IEEE Computer Society, 2010, pp. 77–88.

[4] S. Kanev, T. M. Jones, G.-Y. Wei, D. Brooks, and V. J. Reddi, "Measuring code optimization impact on voltage noise," *Workshop in Silicon Errors – System Effects (SELSE)*, 2013.

[5] T. N. Miller, R. Thomas, X. Pan, and R. Teodorescu, "Vrsync: Characterizing and eliminating synchronization-induced voltage emergencies in many-core processors," in *Proceedings of the 39th Annual International Symposium on Computer Architecture*, ser. ISCA '12. Washington, DC, USA: IEEE Computer Society, 2012, pp. 249–260. [Online]. Available: http://dl.acm.org/citation.cfm?id=2337159.2337188

[6] J.-G. Lee, E. Jung, and D.-W. Lee, "Asymptotic performance analysis and optimization of resource-constrained multi-core architectures," in *Microelectronics, 2008. ICM 2008. International Conference on*. IEEE, 2008, pp. 462–465.

[7] J. D. Davis, J. Laudon, and K. Olukotun, "Maximizing cmp throughput with mediocre cores," in *Proceedings of the 14th International Conference on Parallel Architectures and Compilation Techniques*, ser. PACT '05. Washington, DC, USA: IEEE Computer Society, 2005, pp. 51–62.

[8] M. D. Powell and T. N. Vijaykumar, "Exploiting resonant behavior to reduce inductive noise," in *Proceedings of the 31st Annual International Symposium on Computer Architecture*, ser. ISCA '04. Washington, DC, USA: IEEE Computer Society, 2004, pp. 288–.

[9] M. Gupta, K. Rangan, M. Smith, G.-Y. Wei, and D. Brooks, "Towards a software approach to mitigate voltage emergencies," in *Low Power Electronics and Design (ISLPED), 2007 ACM/IEEE International Symposium on*, Aug 2007, pp. 123–128.

[10] M. S. Gupta, K. K. Rangan, M. D. Smith, G.-Y. Wei, and D. M. Brooks, "Decor: A delayed commit and rollback mechanism for handling inductive noise in processors." in *HPCA*. IEEE Computer Society, 2008, pp. 381–392. [Online]. Available: http://dblp.uni-trier.de/db/conf/hpca/hpca2008.html#GuptaRSWB08

[11] V. J. Reddi, M. S. Gupta, G. Holloway, G. yeon Wei, M. D. Smith, and D. Brooks, "Voltage emergency prediction: Using signatures to reduce operating margins," in *In HPCA'09*, 2009, pp. 18–29.

[12] A. Patel, F. Afram, S. Chen, and K. Ghose, "MARSSx86: A Full System Simulator for x86 CPUs," in *Design Automation Conference 2011 (DAC'11)*, 2011.

[13] S. Li, J. H. Ahn, R. D. Strong, J. B. Brockman, D. M. Tullsen, and N. P. Jouppi, "Mcpat: An integrated power, area, and timing modeling framework for multicore and manycore architectures," in *Proceedings of the 42Nd Annual IEEE/ACM International Symposium on Microarchitecture*, ser. MICRO 42. New York, NY, USA: ACM, 2009, pp. 469–480.

[14] A. Garg, "Characterizing voltage noise in big, small and single-isa heterogeneous cores," Master's thesis, University of Texas at Austin, 2013.

[15] R. Joseph, D. Brooks, and M. Martonosi, "Control techniques to eliminate voltage emergencies in high performance processors," in *Proceedings of the 9th International Symposium on High-Performance Computer Architecture*, ser. HPCA '03. Washington, DC, USA: IEEE Computer Society, 2003, pp. 79–. [Online]. Available: http://dl.acm.org/citation.cfm?id=822080.822802

[16] M. S. Gupta, J. L. Oatley, R. Joseph, G.-Y. Wei, and D. M. Brooks, "Understanding voltage variations in chip multiprocessors using a distributed power-delivery network," in *Proceedings of the Conference on Design, Automation and Test in Europe*, ser. DATE '07. San Jose, CA, USA: EDA Consortium, 2007, pp. 624–629. [Online]. Available: http://dl.acm.org/citation.cfm?id=1266366.1266498

[17] C. Bienia, S. Kumar, J. P. Singh, and K. Li, "The parsec benchmark suite: Characterization and architectural implications," in *Proceedings of the 17th International Conference on Parallel Architectures and Compilation Techniques*, ser. PACT '08. New York, NY, USA: ACM, 2008, pp. 72–81.