Adaptively Tolerate Power-Gating-Induced Power/Ground Noise under Process Variations

Zhe Wang, Xuan Wang, Jiang Xu, Xiaowen Wu, Zhehui Wang, Peng Yang, Luan H. K. Duong,
Haoran Li, Rafael K. V. Maeda, Zhifei Wang
The Hong Kong University of Science and Technology
Email: {zhe.wang, jiang.xu}@ust.hk

***Abstract*—Power gating is one of the most effective techniques to reduce the leakage power in multiprocessor system-on-chips (MPSoCs). However, the power-mode transition during the power gating period of an individual processing unit will introduce serious power/ground (P/G) noise to the neighboring processing units. As technology scales, the P/G noise problem becomes a severe reliability threat to MPSoCs. At the same time, the increasing manufacturing process variations also bring uncertainties to the P/G noise problem and make it difficult to predict and deal with. In order to address this problem, for the first time, this paper analyzes the power-gating-induced P/G noise in the presence of process variations, and proposes a hardware-software collaborated online method to adaptively protect processing units from P/G noise. Sensor network-on-chip (SENoC) is used to gather noise information and coordinate different system components. Meanwhile an online software-based algorithm is developed to effectively decide the noise impact range and arrange protections for affected processing units based on the collected information. We evaluate the proposed method through Monte Carlo simulations on a NoC-based MPSoC platform. The experimental results show that for a set of real applications, our method achieves on average 13.2% overall performance improvement and 13.3% system energy reduction compared with the traditional stop-go method.**

***Index Terms*—Multiprocessor system-on-chip, sensor network-on-chip, reliability, process variation, power gating.**

I. Introduction

Technology scaling allows us to integrate more functional units on a single chip. Current sub-50-nm technologies make it possible to build MPSoCs with up to hundreds of processing units (PUs). However, the sub-50-nm technologies also introduce various sources of non-idealities, such as subthreshold leakage, manufacturing process variations, power supply noise and soft errors in functional blocks. All these issues have posed significant challenges to system designers and chip manufacturers.

Subthreshold leakage is an important challenge for MPSoCs which are generally subject to tight power budgets. It has been revealed that leakage current contributed about 50% of total on-chip currents in sub-100-nm technologies [1]. To this end, power gating has been widely used as an efficient low-power technique to reduce leakage power consumption. In the power delivery network (PDN) of MPSoC, the power gating of each PU can be controlled individually. However, during the power-gating mode transition of a PU, P/G noise will be induced that may affect the normal execution of surrounding PUs. As technology scales down, the continuous reduction in supply voltage levels (lower than 1.0 V for 45-nm technology) makes the P/G noise a serious threat to the functionality and performance of the system.

To alleviate the P/G noise [1] impact, large amount of research efforts have been devoted. [2] proposed a sleep transistor sizing strategy to reduce the worst-case noise. [3] added intermediate voltage levels which allow the state-transition to be made gradually. [4] proposed a GA-based approach to schedule the wake-up order of different sleep transistors in the same power domain to mitigate the P/G noise. There are some other works trying to solve this problem through system-level design method. [5] proposed a power-gating aware task mapping and scheduling strategy for MPSoCs to diminish the impact of P/G noise to system performance. [6] took advantage of on-chip caches as decap to alleviate the P/G noise impact. These works have effectively reduced the impact of P/G noise. However, they did not consider the effects of process variations on the P/G noise.

The increasing manufacturing process variations have become another important challenge imposed by technology scaling. Variabilities widely exist in different aspects of electronic devices, including material, physical geometry, doping and profile [7]. Regarding the interconnect in the PDN, process variations affect metal wire dimensions, such as wire width, wire thickness and dielectric thickness. And for CMOS devices, such as sleep transistor in the PDN, variations lead to electrical parameters deviation from the nominal values, such as threshold voltages, parasitic conductance and capacitance. Since the voltage drop (IR drop, L(di/dt) drop) induced by power gating is highly related to the electrical characteristics of the PDN, the impact of process variation can not be neglected in the analysis of P/G noise. [8] and [9] proposed power gating schemes under process variations. However [8] considered the impact of frequency variations to the proposed corner detectors rather than the variations of P/G noise itself, and [9] focused on the power consumption variations resulted from process variations.

To our best knowledge, this is the first work that explicitly involves the impact of process variations into P/G noise tolerant design. We build a spice-based variational PDN model, and comprehensively analyze the P/G noise charac-

This work is partially supported by GRF620911 and FSGRF15EG04.

[1] Without specific description, P/G noise refers to power-gating-induced P/G noise in the rest of the paper

teristics under process variations, including its spatial and temporal distributions under different variation degrees. The maximum and minimum sample results are shown to illustrate the significant variances of P/G noise distributions under the same power gating scenario. Based on the analysis, we propose a system-level hardware-software collaborated method to alleviate the P/G noise impact on system performance. We take fully advantage of the sensor network-on-chip (SENoC) infrastructure, and develop a P/G noise alleviation protocol to facilitate information collection and resource arrangement. A software-based system is developed to adaptively decide the P/G noise impact range and make effective protections for impacted PUs accordingly. We run Monte Carlo simulations for 6 realistic applications on a 64-PU NoC-based MPSoC platform to evaluate our method. The results show that regarding the mean value, the proposed method achieves on average 13.2% overall performance improvement and 13.3% overall system energy reduction compared with the traditional stop-go strategy.

II. Power Gating induced P/G noise under process variation

In this section, we characterize the effects of process variations on P/G noise. We build Hspice model for the PDN of MPSoC to simulate the power-gating process and profile the generation and propagation of the induced P/G noise. Based on this model, traditional Monte Carlo simulations are conducted to capture the overall influence of different process variables. We will firstly introduce the PDN model with process variations, and then analyze the results to show the necessity of considering process variations in P/G noise tolerant design.

A. Variational Power Delivery Network Modeling

We consider the PDN of a 64-PU MPSoC with homogeneous PU architecture. The PDN consists of a VDD grid and a GND grid. Both of the two grids are modeled as the mesh-based structure. Part of the mesh PDN schematic is shown in the left side of Fig. 1. All the interconnect wires are modeled as resistance (R_w) and inductance (L_w). PUs are modeled as capacitance connected to the virtual V_{dd} nodes and GND grids. Header sleep transistor (M_s) is used for mode transition in power gating process, and it is connected between the VDD grid and the lumped capacitance (C_l) of PU per node. The VDD (GND) grid of each PU contains 576 nodes, and there are totally 36864 (576x64) nodes in the grid of the MPSoC. External power sources are connected to the PDN through package pins which are modeled as resistance and inductance connected serially. This model is compatible with existing works on PDN analysis [10] [5]. We target 32nm technology node, and used the PTM 32nm CMOS models [11] for simulations and design. The nominal supply voltage is set as 0.9V, and the nominal values of PDN parameters are derived from the interconnect model of PTM.

We consider the PDN variations which are resulted from manufacturing process. The sources of variations can be

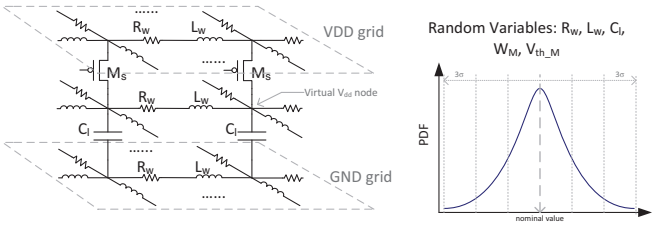


Fig. 1. spice model of the variational power delivery network

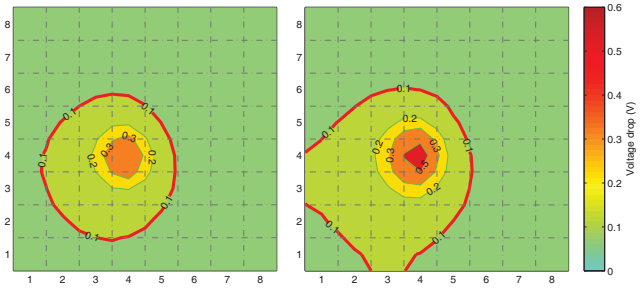
divided into front-end and back-end. Regarding the PDN, the variations in the front-end phase of fabrication process mainly affect the definition of transistor devices, such as the sleep transistors, and the variations in back-end process will lead to interconnect wire imperfections. In the manufacturing process, variations make the parameters of interconnect components and devices deviated from the nominal values. These variations exist among different dies as well as inside a single die. Thus the parameters should be modeled as random variables over different dies and different locations within a die. The 3σ variations of interconnect parasitics of the power grid have been reported to be 30% to 35% of the nominal values by [12]. According to [13], the 3σ variations for transistor L_{eff} is up to 12%, and for V_{th} can be as high as 40% [14]. In this work, we consider the PDN variations in wire metal parasitics, sleep transistor width, threshold voltage. The variations in actual PU logic gates are also considered by involving the variations in the load capacitance. An overview of the overall variational PDN model is shown in Fig. 1.

B. P/G noise analysis under process variation

Only the powering-on phase of a PU will induce significant P/G noise in the PDN with header sleep transistors. We simulate the powering-on process of a single PU, which is called an attacker, and measure the voltage drops of the other PUs. According to [5], the maximum voltage drop that a PU can tolerate without affecting normal execution is 100mV. The number of PUs whose voltage drops violate this threshold is defined as the impact range of the attacker. We characterize the P/G noise in terms of the impact range, maximum voltage drop and voltage recovery time. The random variables are subject to within-die Gaussian distributions [7].

For space limitation, we only show the results of the scenario where PU(4,4) is the attacker for maximum 3σ variations of 12% and 30%. Results of different sample dies are obtained by Hspice Monte Carlo simulations.

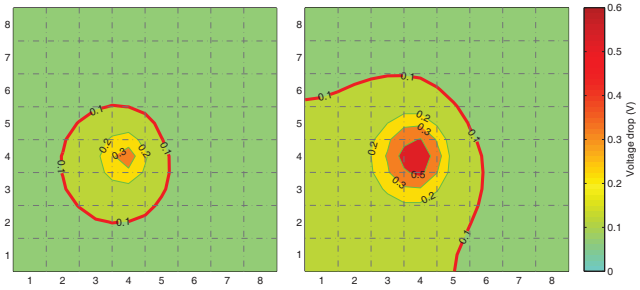
In order to reveal the significant P/G noise variances induced by process variations, we show the results of the largest impact range and the smallest impact range among all sample dies under the same attacker scenario. Fig. 2 shows the P/G noise spatial distribution with $3\sigma = 12\%$. The color grades denote maximum voltage drops at PDN nodes over the whole powering-on period. We highlight the boundary line in red bold format which represents the maximum safe voltage drop, and the area within the line is in danger of malfunctioning. From the figure, the largest impact range is 24 while the



(a) Smallest impact range (b) Largest impact range

Fig. 2. P/G noise distributions induced by attacker PU(4,4) with $3\sigma = 12\%$

smallest range is only 19. Regarding the maximum voltage drop for all impacted PUs (excluding the attacker), the largest value in Fig. 2(b) is over 450 mV and that in Fig. 2(a) is around 330mV. The variance and severeness becomes more significant for $3\sigma = 30\%$. In Fig. 3, the impact ranges could differs by up to 21 PUs between the two sample dies. And the difference between the largest values over all impacted PUs in the two sample dies is more than 260mV.



(a) Smallest impact range (b) Largest impact range

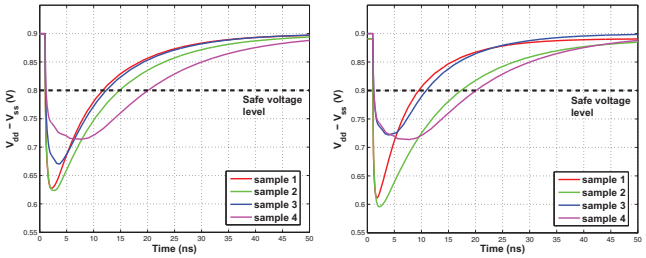
Fig. 3. P/G noise distributions induced by attacker PU(4,4) with $3\sigma = 30\%$

To measure the variance of voltage recovery time, we select four sample dies, and show the supply voltage waveforms of a random PDN node (virtual V_{dd} node) in PU(4,5) which is affected during the powering-on process. Fig. 4 shows the waveforms of the same node on 4 sample dies. We define the recovery time as the period when the supply voltage is below the safe level. From the figure, the longest recovery time could be twice the shortest one in the randomly selected 4 samples under both 3σ variation degrees.

Due to the great runtime of spice simulations, we only run 500 Monte Carlo simulations for each case. Nevertheless, based on this small sample space, we can already see the significant influence of process variations on P/G noise. It is reasonable to infer that the severeness of the P/G noise variation could be even greater in general cases. So based on the results, it is necessary to consider the effect of process variations when trying to address the P/G noise problems.

III. HW-SW COLLABORATED METHOD FOR P/G NOISE TOLERANCE

In order to handle the randomness of process-variation, statistical design and analysis techniques are commonly applied. Yield-driven design and worst-case based design are two



(a) $3\sigma = 12\%$ (b) $3\sigma = 30\%$

Fig. 4. PU(4,5) voltage waveforms measured in 4 samples under attacker PU(4,4)

popular statistical approaches to address the process-variation related problems. The former one tries to satisfy the design requirements of as many variational instances as possible, while the later one finds the worst case scenario first and designs to meet its constraints. In this paper, we seek an alternate deterministic control paradigm to adaptively detect and alleviate the process variation impacts.

We develop a system-level hardware/software collaborated approach to alleviate the impact of P/G noise under process variations. Our method is able to online adaptively decide the protecting range and efficiently protect only the impacted PUs instead of stopping all the PUs, which is the traditional strategy. Using the online detecting method allows the system to accurately decide the impact of process variations for each individual chip. SENoC is used for information collection and power gating arrangement. Meanwhile the software system conservatively decide whether protection is required for each PU based on past noise information, and protects the corresponding PUs before the attacker powering on. Through this mechanism, only "just-right" number of impacted PUs will be protected so that less system degradation will be caused compared to the traditional strategy. In the following parts, we will introduce the SENoC-based infrastructure and the software system respectively.

A. SENoC-based P/G noise tolerant architecture for MPSoC

On-chip sensors have been integrated in recent multiprocessor chips, such as Intel Montecito, AMD Opteron and IBM Cell. They are widely used for online estimation of temperature, power, supply voltage and performance of each processor. [15] proposed a voltage monitor with high sensitivity and accuracy, which can achieve a 10mV DC-resolution and measure 100ps wide spikes. SENoC is an efficient control system prototype for sensor data collection and system coordination. We use the SENoC infrastructure to collect P/G noise information and control the power-gating process. [16] proposed a similar architecture, called MNoC. However, the NoC in our work is shared by PUs and sensors, rather than dedicated for sensors as proposed in [16].

SENoC is composed of on-chip sensors, network agents and a centralized power-gating controller (PGC), and all these components communicate with each other through the NoC. Each PU has a network agent which is in charge of collecting sensor data and making data transactions with the router in the

NoC. The PGC dynamically analyzes the noise information and makes system-level decisions accordingly to protect the system from P/G noise. The PGC can be implemented as a ASIC module in the system.

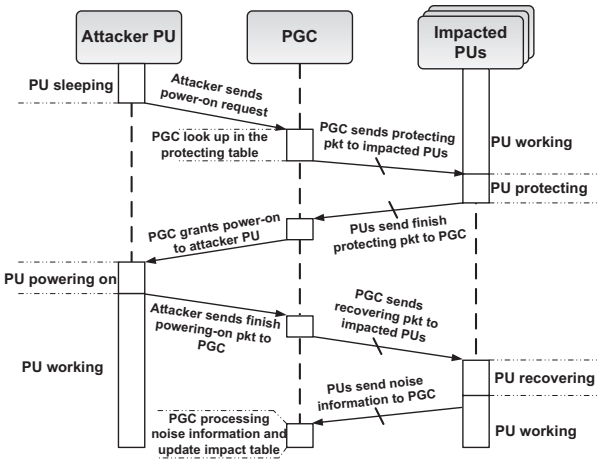


Fig. 5. Timeline sequence diagram for communications among PUs and PGC

We develop a communication protocol for power-gating control. The communication protocol among attackers, PGC and impacted PUs is shown in Fig. 5. The overall work-flow is described as follows. The attacker is originally in sleep mode. When new tasks on it are ready to execute, it will first send power-on request to PGC. Through protecting table look-up, which will be introduced in detail in next subsection, the PGC learns which PUs could be impacted and send protecting commands to them. Once a PU receives the protecting command, it will firstly check its current state. If it is in working state (executing tasks), it will suspend the execution and start protecting. After the PUs are protected, notices will be sent to the PGC. At this time, the attacker is granted to power on. When the supply voltage of the attacker stabilizes, the protected PUs will be told to recover back to the previous working state. For each protected PU, the on-chip sensors monitor the voltage drops for the whole power gating period. After the PU is recovered, the node agent will collect the data from each sensor on the PU and calculate the maximum voltage drop and sent it to PGC for making decisions for PU protecting in the next power-gating event.

All the exchanges of control information are achieved by control packet transmissions in the NoC. In order to reduce the transmitting latency of the control information, these control packets are given a higher priority than the data packets in utilizing network resources. In each PU, the control information is directly packetized and sent out in the node agent without being pushed into the output queue. In the network, control packets have a higher priority to reserve virtual channels and go through switching crossbars. It is important to note that the noise information obtained after power gating is transmitted as normal data packet and it does not have to be sent out right after the recovery of the PU.

B. software system for the decision of PU protecting

The software system is implemented in the PGC. The main task for the software system is to decide whether a PU requires protection under a certain attacker scenario. In this work, we develop a posterior predictive model to conservatively make protecting decisions based on past knowledge. The reason of using predictive strategy is mainly based on the transient nature of P/G noise. The PUs can not make in-time protection when the sensors detect dangerous voltage drops. The protecting actions, such as clock gating and data backup, will take at least tens of clock cycles to finish while the transient voltage drop could have caused malfunctioning. So we protect a PU based on its past noise information which can be collected in previous power-gating events.

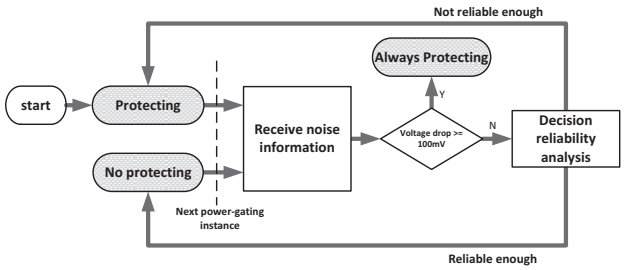


Fig. 6. Decision flow for PU protection

We use a table, called protecting table, to store the protecting decision (*protecting/no-protecting*) for each PU under different attacker scenarios. In the table, each row represents an attacker scenario, and the protecting decisions for all the PUs under this attacker scenario are stored in the corresponding positions of this row. For a MPSoC with n PUs, the table size is n^2. The LUT can be implemented in SRAM. Every time after a power-gating event happens, the PGC will collect the noise information and update the protecting table. So every time during the power-gating period, the PGC only need to look up the protecting table to learn whether a PU requires protection or not.

At the beginning, all the elements in the protecting table are initialized to be *protecting*, which means all PUs under all attacker scenarios will be protected. As power-gating events keep occurring, new decisions will be made based on the noise information (maximum voltage drop) obtained from the power-gating events. For a certain attacker scenario, the protection decision flow for a PU is shown in Fig. 6. In this flow, if the voltage drop of a PU exceeds the 100mV threshold, the PGC will always protect this PU in the future under this attacker scenario. However, if the voltage drop of a PU is reported to be safe for once, we cannot immediately assure that it does not require protection. Because there could be other factors, such as the different surrounding PU modes (sleep/working), causing variations to the voltage drops in different times of power gating events which are yet under the same attacker scenario. So we need a reliability analysis to cautiously make protecting decisions. We use past voltage drops as samples to estimate the voltage drop distribution of

general situations, and dynamically test whether the current estimation is credible enough to make *no-protecting* decision. So a key problem is how many samples do we need in order to make estimations with a certain confidence level. To answer this question, we model and solve the problem as follows.

Assuming that N voltage drop values v_i have been recorded as samples, the mean value of general-case voltage drop V can be approximated by

$$\tilde{V} = \frac{1}{N}\sum_{i=1}^{N} v_i \quad (1)$$

Based on the statistical inference theory, we can tell that the $\tilde{V}$ lies in a certain range around V with some confidence level. We want to find the relationship between the confidence level and the value of N. Provided that the variance, denoted as σ^2, of the distribution of V is finite, the central-limit theorem is applied and we have

$$Pr\{|\tilde{V} - V| < \frac{\beta\sigma}{\sqrt{N}}\} \approx \Phi(\beta) - \Phi(-\beta) \quad (2)$$

where $\beta\sigma/\sqrt{N}$ is the half length of the confidence interval ξ, σ^2 is the variance of the distribution of V, and $(\Phi(\beta) - \Phi(\beta))$ is the corresponding confidence level that V lies in $(\tilde{V} - \xi, \tilde{V} + \xi)$. The function $\Phi()$ is the CDF of unit normal distribution. In order to keep the maximum 3σ value of V less than the threshold voltage drop V_t. ξ should satisfy the following relationship.

$$\xi = \frac{\beta\sigma}{\sqrt{N}} \leq V_t - \tilde{V} - 3\sigma \quad (3)$$

However, σ is hard to derive by theoretical analysis. Based on [17], we can approximate σ^2 as below,

$$\sigma^2 \approx \frac{N}{N-1}\tilde{\sigma}^2 = \frac{N}{N-1}(\frac{1}{N}\sum_{i=1}^{N} v_i^2 - \tilde{V}^2) \quad (4)$$

Substituting equation 4 into equation 3, we get,

$$\frac{(\beta + 3\sqrt{N})\tilde{\sigma}}{\sqrt{N-1}} \leq V_t - \tilde{V} \quad (5)$$

In Equation 5, the value of $\tilde{V}$ and $\tilde{\sigma}$ can be calculated dynamically when new noise information is received. Provided the confidence level required (β value), every time new noise information sample is received, the PGC will calculate the value of $\tilde{V}$ and $\tilde{\sigma}$, and then check whether (5) is satisfied. If it is satisfied, then we update the PU protection decision to be *no-protection*. Even though *no-protection* decision is made, according to Fig. 6, the noise information will be continuously collected for reliability analysis. It is important to note that in Equation 4, the credibility of the approximation of σ also depends on the sample size. So in our algorithm, at least a certain number of samples are required before start to make decisions, and we call the number as *basis sample size*.

The main overhead of this algorithm is the memory space to store the protection information for each PU under different power gating scenarios. For the worst case, the memory overhead is $O(n^2)$ where n is the number of PUs. The computation runtime overhead is negligible with constant value time complexity.

IV. Experimental results

We implement the proposed hardware-software collaborated method on a 64-PU MPSoC system with mesh-based NoC structure. The frequencies of all the components are set as 1 GHz. For the NoC, XY routing and wormhole switching protocol are applied, a fixed packet size of 8 flits with 32 bits per flit (same as the width of each router port) is assumed. The routers are working in a pipelined manner and we assume three cycles delay for flits to cross a router. Link delay is confined in one cycle. These settings are compatible with common NoC studies. The power model is derived from [6]. The time required for a PU to enter/leave the protection mode are both 10ns [5]. We assume that the PGC have the same power consumption as the PUs in working state. The whole system is built on a SystemC based platform, and detailed cycle-level simulations are conducted. The variational P/G noise model is derived based on the spice simulation. We select 6 realistic applications, namely *FFT-1024_complex, Sparse, Fpppp, Robot, MolecularDynamics (MD)* and *LDPC* from the COSMIC benchmark suite [18] and execute them on the MPSoC for performance evaluation. We adopted the traditional idle time-based power gating schedule, where the PUs that are idle for a certain time will ask for power-gating.

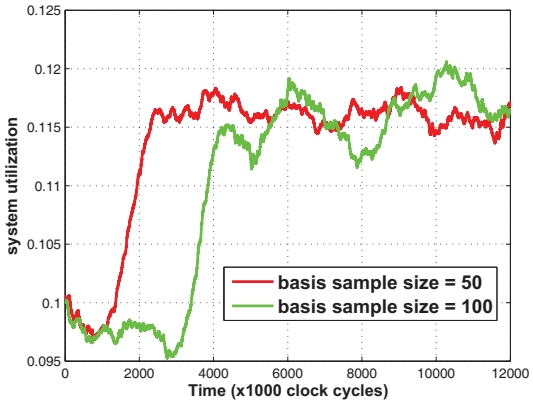


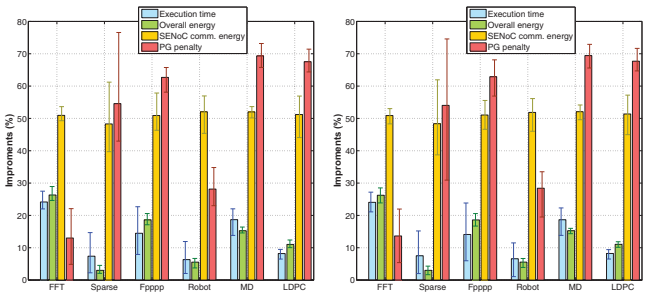
Fig. 7. Transient system utilization for *FFT-1024_complex* under different basis sample sizes

We firstly use the metric of *transient system utilization* to show the process of online impact range decision. *transient system utilization* is defined as the ratio between the number of PUs that are executing tasks and the total PU number. The application is executed iteratively and the measure step-size is 1000 clock cycles. We show the results of *FFT-1024_complex* in Fig. 7.

From Fig. 7, it is easy to observe that the utilization gradually increases as time goes on. As the system runs, power-gating events keep occurring. Meanwhile the PGC collects the P/G noise information of different PUs. The protecting decisions for PUs are made as the sample spaces get enlarged.

With more and more protecting decisions made for different PUs, the impact range shrinks to a stable area. The system utilization get increased as more safe PUs do not need to stop working when there is power-gating happening. And finally, the system reaches a stable stage. For different *basis sample sizes*, the time needed to make decisions will be different. The larger the basis sample size is, the longer time is needed for the system to reach the stable stage.

We also run Monte Carlo simulations for all the 6 realistic applications to evaluate our method. Different metrics are used for evaluation, including application execution time, overall system power consumption, SENoC communication energy consumption and power-gating induced performance penalty which is defined as the number of cycles that the PUs spend for P/G noise protection. We compared our method to the traditional *stop-go* method [6] where all the PUs are protected when power-gating happens, and performance improvements at system stable stage are shown in Fig. 8.



(a) $3\sigma = 12\%$ (b) $3\sigma = 30\%$

Fig. 8. Percentage of performance improvements compared to *stop-go* strategy

In Fig. 8, the height of each bar refers to the mean value and the upper/lower edges of the vertical line segment denote the maximum/minimum values of all the samples. Obviously our method outperforms the *stop-go* method in terms of all four metrics. Regarding the mean values, our method reduces on average 13.2% execution time, 13.3% energy consumption, 50.9% SENoC communication energy consumption and 49.2% power-gating penalty. The improvements for execution time and power-gating penalty are resulted from the less PUs being protected, and the improvements are relatively small for the applications with small number of tasks but long execution time for each task. The reduction in energy consumption is mainly due to fewer power-gating control packets sent.

V. Conclusion and Future Work

In this paper, for the first time, we analyze the effect of process variations on the power-gating-induced P/G noise. By developing a spice-based variational power delivery network model for MPSoCs, we conduct Monte Carlo simulations to exhibit the significant variance of P/G noise imposed by process variations. Simulation results show the necessity of considering process variations in alleviating the impact of P/G noise. Based on the analysis, we propose a hardware-software collaborated method to adaptively protect processing units from P/G noise. Sensor network-on-chip (SENoC) is used to gather noise information and coordinate different system components. Meanwhile an online software-based algorithm is developed to conservatively decide the P/G noise impact range and effectively arrange protections for each PU based on the collected information. We evaluate the proposed method through Monte Carlo simulations on a NoC-based MPSoC platform, and the experimental results show that our method achieves significant improvement on performance and energy consumption. For future work, the impact of process variations to NoC components and sensor functionality could be investigated. And error handling mechanism could be developed to deal with prediction fails.

References

[1] I. Ferzli and F. Najm, "Statistical verification of power grids considering process-induced leakage current variations," in *Computer Aided Design, 2003. ICCAD-2003. International Conference on*, Nov 2003, pp. 770–777.

[2] J. Gu, H. Eom, J. Keane, and C. Kim, "Sleep transistor sizing and adaptive control for supply noise minimization considering resonance," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, vol. 17, no. 9, pp. 1203–1211, Sept 2009.

[3] M. Chowdhury, J. Gjanci, and P. Khaled, "Controlling ground bounce noise in power gating scheme for system-on-a-chip," in *Symposium on VLSI, 2008. ISVLSI '08. IEEE Computer Society Annual*, April 2008, pp. 437–440.

[4] H. Jiang and M. Marek-Sadowska, "Power gating scheduling for power/ground noise reduction," in *Design Automation Conference, 2008. DAC 2008. 45th ACM/IEEE*, June 2008, pp. 980–985.

[5] Y. Wang *et al.*, "Power gating aware task scheduling in mpsoc," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, vol. 19, no. 10, pp. 1801–1812, Oct 2011.

[6] X. Wang *et al.*, "Active power-gating-induced power/ground noise alleviation using parasitic capacitance of on-chip memories," in *Design, Automation Test in Europe Conference Exhibition (DATE), 2013*, March 2013, pp. 1221–1224.

[7] P. Ghanta and S. Vrudhula, "Analysis of power supply noise in the presence of process variations," *Design Test of Computers, IEEE*, vol. 24, no. 3, pp. 256–266, May 2007.

[8] C. Yeh, Y.-C. Chen, and J.-S. Wang, "Towards process variation-aware power gating," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, vol. 20, no. 11, pp. 1929–1937, Nov 2012.

[9] D. Kannan *et al.*, "Temperature and process variations aware power gating of functional units," in *VLSI Design, 2008. VLSID 2008. 21st International Conference on*, Jan 2008, pp. 515–520.

[10] Gupta *et al.*, "Understanding voltage variations in chip multiprocessors using a distributed power-delivery network," in *Design, Automation Test in Europe Conference Exhibition, 2007. DATE '07*, April 2007, pp. 1–6.

[11] Nanoscale integration and modeling (nimo) group. predictive technology model (ptm), http://ptm.asu.edu/.

[12] D. S. Boning and S. Nassif, "Models of process variations in device and interconnect," in *Design of High Performance Microprocessor Circuits, chapter 6*. IEEE Press, 1999.

[13] International Technology Roadmap for Semiconductors, 2013 edition.

[14] C. Hernandez *et al.*, "Characterizing the impact of process variation on 45 nm noc-based cmps," *Journal of Parallel and Distributed Computing*, vol. 71, no. 5.

[15] V. Petrescu, M. Pelgrom, H. Veendrick, P. Pavithran, and J. Wieling, "Monitors for a signal integrity measurement system," in *Solid-State Circuits Conference, 2006. ESSCIRC 2006. Proceedings of the 32nd European*, Sept 2006, pp. 122–125.

[16] J. Zhao *et al.*, "A dedicated monitoring infrastructure for multicore processors," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, vol. 19, no. 6, pp. 1011–1022, June 2011.

[17] P. Jackel, *Monte Carlo Methods in Finance*.

[18] Z. Wang *et al.*, "A Case Study on the Communication and Computation Behaviors of Real Applications in NoC-based MPSoCs," in *IEEE Computer Society Annual Symp. VLSI*, 2014.