Impact of Process and Temperature Variations on Network-on-Chip Design Exploration

Abstract

With the continuing scaling of CMOS technologies, process variation is becoming a key factor highly impacting system-level power and temperature. Traditional methods of assuming a uniform temperature and no process variation can lead to gross inaccuracies even for system-level design, thus it is critical to consider the effects of process variation and temperature variation during early design exploration. In this paper, we describe the implementation of an architecture-level early-stage design space exploration tool that incorporates the effect of process and temperature variation for Network-on-chips(NoC). The tool is used to study the impact of process and temperature variations on power and energy-delay-product-per-flit metrics for different NoC architectures, and our simulation results show that design choices are very much influenced by the effects of process and temperature variation, thus demonstrating the importance of considering, and enabling the high-level impact analysis of process and temperature variation early in the design flow.

1. Introduction

The continuing scaling of technology poses several design challenges for future high performance architectures. One such challenge is dealing with the increased effects of process variation. Besides, the continuous increase in operating frequency along with higher on-chip integration of functionalities has been exacerbating chip power density and within-die temperature fluctuations.

These problems and their impact on power and performance have gained attention in the past few years. Techniques that mitigate process variations have been proposed [1, 2]. Several works used actual chip measurements to study the magnitude and spatial correlation of process variation [3, 4], while others developed statistical models for estimating leakage power and yield in the face of process variations [5–10]. However, power estimation accounting for variability at architecture or system level have gained attention only recently [11–16]. Many decisions taken at system level can significantly affect the power and temperature profile as well as the overall performance. Assuming a uniform temperature and no process variation can lead to substantial inaccuracies in system-level design choices. It is thus important to accurately and efficiently estimate power taking into account process and temperature variations so that they can be accounted for early in the design stage.

In this paper, we investigate how system-level design decisions for multi-core chips are affected by process and temperature variations, focusing specifically on the network-on-chip (NoC) interconnecting these chips, as NoCs are emerging as the scalable interconnect replacing buses and crossbars in many-core chips [17, 18]. Except for [14], prior works on system level power estimation were performed for small circuit functions or uniprocessors [11, 13, 15, 16]. NoC power estimation is important as NoCs consume a significant amount of total on-chip power. For example, MIT RAW's on-chip interconnection network consumes 36% of total power [19], Intel 80-core Polaris's on-chip network consumes 39% of total power [20].

In this paper, we extend Polaris [21], a system level power/performance design space exploration tool for NoCs to take into account process and temperature variations. In particular, we focus on leakage power estimation as dynamic power is not significantly impacted by channel length variations and threshold voltage variations [13]. Polaris is a rapid design space exploration tool that estimates network power-performance and provides designers with relative power-performance rankings that can be used to make architectural choices at early design stage. By factoring in process and temperature variations, our results show the importance of considering and quickly estimating the impacts of both the process and temperature variations together because the design decisions taken at early design stage might be changed.

The remainder of the paper is organized as follows. Section 2 presents background on how process and temperature variations impact leakage power. Section 3 explains how Polaris is extended to take into account process and temperature variations. Section 4 studies the impact of process and temperature variation on on-chip interconnection network power and energy-delay-product-per-flit metrics estimation across different architectures. The paper is then concluded in section 5.

2. Background and Motivation

Systematic and random variations in process, supply voltage and temperature have become a major challenge to future high performance architecture design. Process variation is mainly caused by the difficulties in the precise control of lithography and inherent random processes, thus causing line edge roughness. With the continuing reduction of the number of dopant atoms in the channel between source and drain in a MOSFET, random dopant-density fluctuation causes variation in the transistor characteristics. Moreover, the increase in operating frequency for each processor generation has resulted in significantly higher power-density and on-die temperature; additionally the spatial and temporal variation in workloads and levels of activity across the multi-core die, the uneven thermal environment of the

chip, etc. inevitably lead to subtantial power-density and temperature variation within the chip. Temperature variation can lead to hot-spots and points of reliability failure and has an exponential realationship with leakage which can cause further exacerbation of power [22].

Process variations can be classified into die-to-die (D2D) and within-die (WID) variations. D2D variations affect all the transistors on the same chip in the same way while WID variations affect different transistors differently on the same chip [5]. For higher-performance IC designs, D2D and WID variations have a significant impact on system performance and power consumption. A key process parameter subject to variation is the transistor threshold voltage (V_{th}). Process and temperature variation leads to variation in threshold voltage, and leakage current has exponential relationship with threshold voltage. So transistor sub-threshold leakage is strongly dependent on threshold voltage and temperature. Variation in functionality across the chip results in uneven power dissipation; this variation results in uneven temperature profile across the chip, which in turn causes leakage variation across the chip and increases the variability.

The sub-threshold leakage current can be given by [23] as:

$$I_{sub} = I_0[1 - exp(-\frac{V_{ds}}{V_t})]exp(\frac{V_{gs} - V_{th} - V'_{off}}{nV_t}) \quad (1)$$

$$I_0 = \mu \frac{W}{L} \sqrt{\frac{q\varepsilon_{si} \cdot NDEP}{2\Phi_s}} V_t^2 \quad (2)$$

where V_{th} is the threshold voltage, V_{ds} is drain-to-source voltage, V_{gs} is gate-to-source voltage, and $V_t = \frac{kT}{q}$.

From Equation (1) and Equation (2), we can get that:

$$I_{sub} \propto T^2 exp(-\frac{qV_{th}}{nkT}) \quad (3)$$

Equation (3) illustrates that I_{sub} has strong relationship with temperature T and threshold voltage V_{th}. As V_{th} is highly impacted by process variations due to L_{eff} variation and dopant concentration variations, I_{sub} will also be critically impacted.

Figure 1 shows the relationship between leakage power and threshold voltage V_{th} [23]. The leakage power is normalized to the leakage at fixed V_{th} of 0.25V. We can see leakage power increasing rapidly with the decrease of V_{th}. Fig. 2 plots the leakage power as a function of temperature [24]. The leakage power is normalized to the leakage power at 40°C. We can see that leakage power increases rapidly with the increase in temperature. Besides, different functionality across different circuit blocks will lead to further temperature fluctuation on the chip, thus causing uneven leakage power on the chip.

In short, we see that leakage power is strongly affected by process variation and temperature variation. With the continuous scaling of technology, this problem is becoming more and more critical. We can no longer assume uniform temperature and ignore process variation even at the early stages of design.

3. Proposed Work

Polaris is a system level roadmap for on-chip interconnection networks developed by Soteriou et al. [21, 25]. It

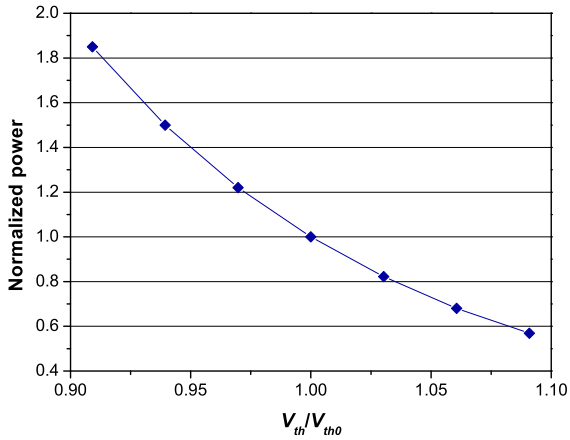

Figure 1. Leakage power and V_{th} relationship

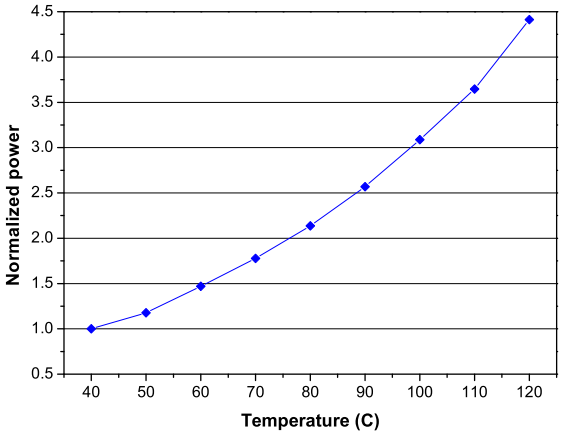

Figure 2. Leakage power and T relationship

is a toolchain that can rapidly explore a large design space of NoCs, estimate system-level power/area/delay and guide designer towards the most suitable architectures that best balance cost/performance based on designer's needs at the early design stage. The reason we choose to use Polaris is because it incorporates a large design space of 7872 NoC designs and the simulation time is much faster than detailed cycle-accurate simulation while ensuring reasonable accuracy. Thus, this tool is suitable for us to use for exploring the effects of process and temperature variations for early design space exploration. We choose to use Polaris also because Polaris and Orion have a significant user base of several hundreds users, and so our extending of it to include process and temperature variations will hopefully benefit those users as well.

We will first introduce the original Polaris toolchain which does not consider parameter variation effects, before explaining how the tool is extended to estimate NoC power with consideration of process variation and temperature fluctuation.

3.1. Introduction to the Polaris toolchain

Figure 3 shows Polaris's flow chart for rapidly estimating power/area/delay of large design space and provide designers with roadmapping tables, i.e. tables with estimates of various metrics for alternative NoC designs at different processes [21]. The flow chart includes three steps. The first

step is synthetic traffic traces generation using Trident [26], which are then used to drive and exercise each NoC design. Trident is a synthetic traffic generator which can generate artificial traffic traces based on designer's requirement. The author of [26] first derives a traffic model that can accurately capture characteristics of real-world traffic, and then classify them into several categories which can represent different traffic traces. The designers can specify their traffic characteristics, and Trident can generate artificial traffic traces according to designer's application. The second step is network resource utilization with LUNA (Link Utilization for Network Analysis) [27]. LUNA is a network analyzer which takes the traffic traces from Trident as input and calculates network resource utilization for a wide range of architectures, and then estimates network activity/utilization, delay and contention. The third step is network power and area estimation using Orion [28]. Orion is a power and area library for different router configurations which projects potential circuit structures for each configuration at each technology node. It takes LUNA's resource utilization information for each router and link as input activity and factors that with Orion's router power estimates and wire capacitance estimates projected by ITRS [29] along with router distance either provided by designer or projected by ITRS [29]. After this, router power and link power are added together to estimate the whole on-chip interconnection network power. Orion also estimates the area for each network-on-chip configuration based on the prescribed circuit structure and assumed layout.

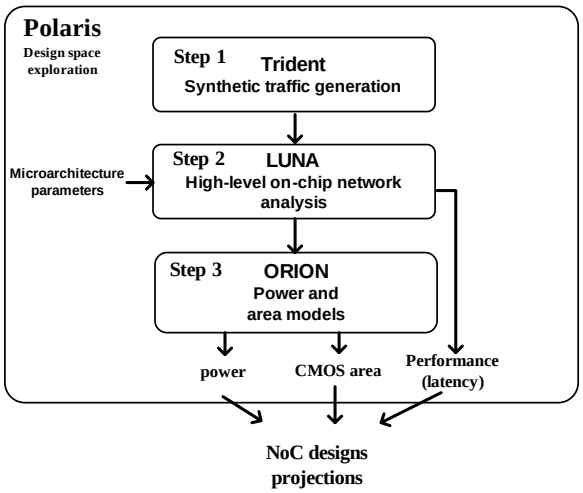

Figure 3. Polaris network roadmapping toolchain

After the three steps above, Polaris compares results of power/area/delay among each network-on-chip configuration and provides the designer with roadmap tables: for instance, network power, latency, energy-delay-product-per-flit and power per area. Based on this roadmap table information, designers can then choose the most suitable network-on-chip architecture that satisfies their requirements and design constraints.

3.2. Extension of Polaris power models to consider process and temperature variations

The original Polaris toolchain explained in 3.1 assumes uniform temperature within the chip and no process variation. To study how process and temperature variation affects the power consumption for on-chip interconnection networks, we extended this tool to calculate power consumption with consideration of process variation and temperature fluctuation for on-chip interconnection networks.

Figure 4 presents an overview of the power modeling methodology considering process and temperature variations which is incorporated into Polaris toolchain. This iterative approach was adapted from [11]. In [11], the author estimates the full chip leakage considering power supply and temperature variations. We use this methodology to consider process and temperature variations. We extended Orion's leakage power model [30] to make it process and temperature aware. For dynamic power, since it is not as significantly affected by process and temperature variations as leakage power, we left Orion's dynamic power model [28] untouched.

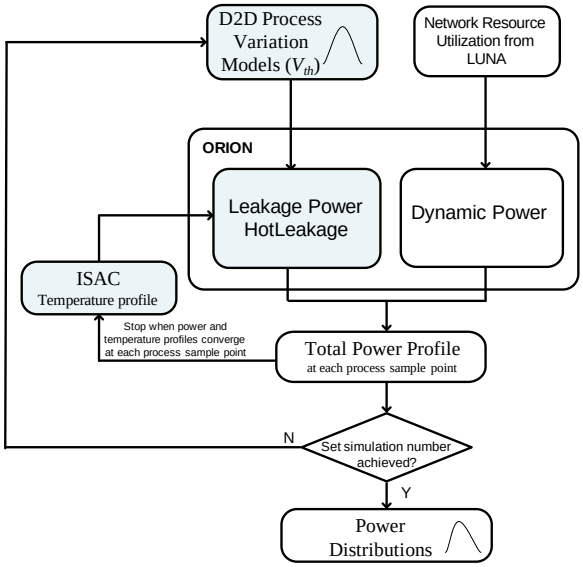

Figure 4. Power model flow chart considering process and temperature variation using Monte Carlo simulation

In Figure 4, a process sample point from the D2D process variation model is fed into the leakage power model, and leakage power consumption is calculated at an initial temperature. Combined with dynamic power estimation, the total power profile is calculated. This power profile is then fed into thermal simulator to estimate the thermal profile. This thermal profile is then used to re-calculate the leakage power profile since leakage power has strong dependence on temperature. This power changes will again affect temperature profile [11]. After several iterations, the difference between each loop will be small enough and we can get acceptable accuracy for the power and temperature estimation with high efficiency. This procedure is performed based on Monte Carlo simulation and randomly generated sample point is used for each simulation until a certain confidence level is ensured. After Monte Carlo simulation, a power distribution profile is generated for this design point. Designer can run the simulation for different design choices and select the most suitable design parameters that satisfy the design requirements. We will next explain each of the new components in detail.

3.3. Leakage power model

As transistor feature size gets smaller and smaller, it's getting more and more difficult to precisely control the fabrication process, thus causing process variations. There are two major parameters subject to process variations that are important for microarchitectural and system level design. One is the effective transistor channel length (L_{eff}), the other is transistor threshold leakage (V_{th}). We choose to use V_{th} as the process variation parameter in our power model, but other sources of variations such as L_{eff} can be incorporated in our work as well.

In [30], the authors propose an architectural level leakage power modeling methodology and the proposed leakage power model was incorporated into Orion. While it is validated against Spice to be accurate at 70nm, the leakage power model does not incorporate the effects of process and temperature variations. Besides, the leakage power model is for sub-threshold leakage. With the contiguous transistor scaling, gate oxide thickness is projected to scale for future technologies [29], which causes gate leakage to increase rapidly. So it's important to consider gate leakage as well in the leakage power model.

In order to consider process variation and temperature variation effects, we incorporate leakage current models from HotLeakage developed by Zhang et al. [31] into Orion for estimating sub-threshold leakage and gate leakage. HotLeakage is an architecture level leakage power model that includes temperature, voltage, gate leakage and parameter variations. It uses the BSIM3 V3.2 [32] leakage equation of MOSFET to model the leakage of a single transistor and extends the Butts-Sohi [33] model to take into account the stack effect and interaction among multiple transistors. It includes parameters of supply voltage, threshold voltages, oxide thickness and temperature.

In our simulator, we replace the original leakage model in Orion with the leakage power models from HotLeakage. We fine tune the sub-threshold leakage current model from HotLeakage with simulation and real chip measurements from [24]. We keep the gate leakage current model from HotLeakage untouched.

While Hotleakage models supply voltage and oxide thickness parameters, we only concentrate on V_{th} and T as our parameters because these two parameters are the major contributors to sub-threshold leakage. For V_{th}, we consider D2D V_{th} variation currently because we are targeting at early design space study and D2D variatons are more interesting to us. We assume the same V_{th} in each chip while different chips have different V_{th}. We also assume a normal distributions among chips for V_{th} variations. We estimate the on-chip temperature at the router granularity, that is different routers on chip have different temperatures based on the traffic activity, but a single router is modeled as a single temperature source.

3.4. Thermal model

To calculate the temperature profile, we use the thermal simulator ISAC developed by Yang et al. [34]. ISAC is a chip-package thermal analysis tool used in IC synthesis and design. It takes a three-dimension chip and package thermal conductivity profile, as well as power dissipation profile as input parameters. It uses a multigrid incremental solver to progressively refine thermal element discretization and rapidly produce a temperature profile. The thermal analysis provided by ISAC is very fast due to their adaptation of spatial resolution and temporally decoupled element time marching techniques. Further, the authors show an accuracy of 99% using industrial and academic synthesis test cases and chip designs. Hence we chose to incorporate this model for thermal profile analysis in our early design stage exploration.

In our simulator, we use steady-state thermal profile. Designer can provide a power profile for the computing and storage parts on the chip. The simulator calculates the power consumption of each router on the chip using Orion with the leakage model from HotLeakage. Then the NoC power profile is combined with the user-defined computing and storage power profile to form a whole chip power profile. This whole chip power profile is then fed into ISAC. ISAC rapidly calculates the temperature for the whole chip and provides the thermal profile for the chip as the output. The updated thermal profile is then fed into Orion to recalculate the power consumption. This procedure will iterate until the temperature and power profile converges.

3.5. Monte Carlo Simulation

We use Monte Carlo simulation to estimate power distribution for interconnection network under process variation. For each simulation, a sample V_{th} is randomly generated from the pre-defined process variation model and leakage power is calculated at this V_{th}. This step is performed with enough sample points to ensure a pre-defined confidence level is achieved, thereby generating a power distribution profile.

3.6. PolarisPT: Polaris toolchain considering process and temperature variations

After explaining each part of the power model flow chart, now we incorporate this power model into Polaris toolchain. Figure 5 presents the new Polaris toolchain (PolarisPT) which now considers process and temperature variations for power estimation. The artificial traffic generator Trident first generates synthetic traffic trace, and this traffic trace is fed into LUNA to capture the network resource utilization and calculate the network latency. The network resource utilization information is then fed into Orion (which now includes a process-variation-aware leakage power model) to calculate the power consumption distribution with consideration of process and temperature variations. Monte Carlo simulation is performed at this step with enough sample points so that a pre-defined confidence level can be achieved. (designers can define the number of samples as simulation input as well). Within each run at this step, leakage power is calculated iteratively with the sampled V_{th} and simulated temperature until the power/thermal profile converged. After the Monte Carlo simulation, a power distribution envelope is generated. Orion also calculates the area for on-chip interconnection networks at this step. Finally, PolarisPT will provide designers simulation results containing network power distribution, latency, area, energy-delay-product-per-flit for different network configurations.

4. Experimental results

This section analyzes the impact of die-to-die(D2D) threshold voltage V_{th} variation and within-die (spatial) temperature variation on interconnect network design. Espe-

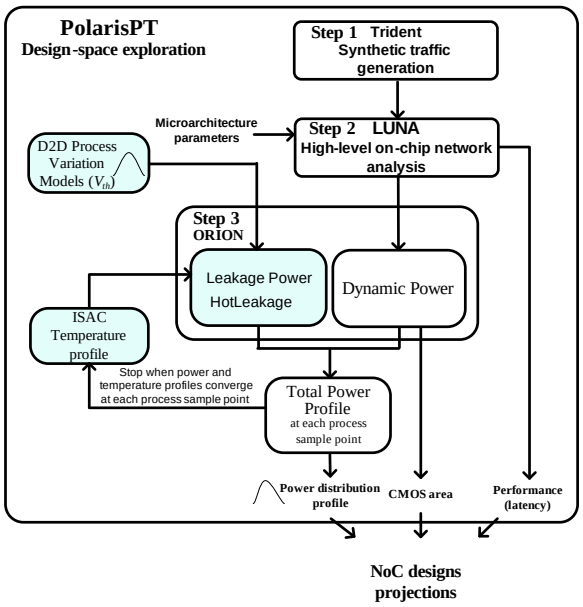

Figure 5. PolarisPT toolchain with consideration of process and temperature variations

cially, we study the power and the energy-delay-product-per-flit (EDPPF) distributions for various interconnection network architectures.

4.1. Experimental setup

We consider a mockup chip similar to Intel's 80-core teraflops NoC [20, 35] for our baseline system and analysis. While this is a hypothetical chip, the parameters are derived largely from the aforementioned 80-core chip for a realistic baseline. As such, our experiments assume 64-node NoC designs, 65nm processing technology, a supply voltage of 1.2V and the operating frequency of 3.8GHz. For our study, we scaled this chip's floorplan to 64 cores, which comes to 14.4mm by 14.4mm. The standard deviation of D2D threshold voltage is assumed to be 6% based on the ITRS projection [29]. Table 1 summarizes the parameters used in our simulation.

Table 1. Parameter values

Technology	65 *nm*
Clock frequency	3.8 GHz
Supply voltage	1.2 *V*
V_{th}	μ = 0.25*V*, σ / μ = 6%
Non communicatin component power	1.2W/core
Confi dence interval	95%
Die size	14.4 *mm* $\times$ 14.4 *mm* $\times$ 0.6 *mm*
Heat sink	60 *mm* $\times$ 60 *mm* $\times$ 6.8 *mm*
Ambient air temperature	45 °C

Our study considers several network topologies: 2D mesh plain, 2D torus plain, 2D mesh with express cube, 2D torus with express cube, 2D mesh with hierarchical link and 2D torus with hierarchical link. Table 2 shows the design space studied in our experiment. For each topology, we assume a packet has five 64-bit flits [1]. Synthetic traffic traces generated from Trident [26] are used as the input to the system simulator. Due to the limited space, the simulation results shown here are for *long-distance, moderately bursty and hot-spot traffic*, as characterized by Trident. Our results show similar trends for other traffic patterns though.

[1] Flit is flow control unit, a fixed-length segment of a packet.

Table 2. Design space explored in the experiment

Topology	2D mesh plain
	2D mesh with express cube interval 2,3
	2D mesh with hierarchical link interval 2,3,4
	2D torus plain
	2D torus with express cube interval 2, 4
	2D torus with hierarchical link interval 2,3,4
Buffer size (64-bit flits)	4, 8, 16, 32
Virtual channels per link	1, 2, 4, 8
Routing	Deterministic routing
Flow control	Wormhole, virtual cut-through

4.2. Effects of process variation and spacial temperature variation

Our baseline experiment involved simulating our "chip" under the assumption of no D2D V_{th} variation and a single uniform temperature of 80 °C. Then the chip power and the energy-delay-product-per-flit for the traffic described above were estimated for the various interconnection network configurations. We then simulated the system considering D2D V_{th} variation and the spatial T variation on a set of 500 chips. For all the described experiments, we made a simplifying assumption that the change in temperature did not affect the power dissipation in components other than the network. In other words, we assumed that power dissipation in storage and computation blocks of the simulated chips all remained constant, and that the leakage of the network components were affected by the temperature and vice versa. (We suspect that the change in power dissipation and performance of these non-network blocks will further accentuate the total effect of variation on the design metrics.) We modeled total leakage as the sum of sub-threshold and gate leakages. Although our system can be easily extended to assign different power densities for the different computation and storage areas of each tile (a physical area associated with a core and its router) on the chip based on block-level power estimations as desired, for our simulation, we used a uniform power density for these non-network components.

In figure 6, the x-axis shows the first eight network architectures that have the lowest power consumptions assuming *no process variation and uniform temperature*. By architecture here we mean the network topologies and their specific configurations denoted in the format: *topology_variant (buffer_size, number_of_virtual_channels)*. For example, by "mesh plain (4,2)" we mean a 2D mesh topology in its *plain* variant with an input buffer-size of 4 flits/virtual-channel and 2 virtual channels per router port. From left to right on the x-axis, the architectures are ordered in increasing value of their estimated power consumption. The y-axis represents the power consumption normalized to the lowest-power NoC architecture, among the set considered (still assuming no variation effects), which is mesh plain with 4 buffers per virtual channel and 1 virtual channel per port (i.e., no virtual channel). The bottom plot in the figure shows two bars for each architecture where the left bar indicates estimated interconnect power assuming no variations. The right bar shows the estimated power when considering

both the effects of V_{th} and T variations. The bands in the right bars show the power variations across the set of chips simulated. The longer bands imply higher variations across chips. From this bottom part of the figure, we can see that when considering the effects of V_{th} and T variations, the average power consumption increases compared to the no variation scenario, for each network architecture shown in the figure. This is because leakage power has exponential relationship with T. The higher the T, the leakier the transistors. Now with the consideration of V_{th} and T variations, the power consumption becomes a distribution rather than a single, deterministic number. We note that the larger the average power consumption, the higher the variations. We have used the *mode* of the distribution to mean the single 'average power' number in the discussion here. (In practice, typically the slowest and the fastest, i.e. very leaky, parts get rejected and the rest get binned. So, the average power may be suitably defined for each bin considered.)

The upper part of the figure 6 shows the product of the power distribution's standard-deviation and mean ($\sigma \times \mu$) – this is a metric relevant while considering parametric yield and total power of chips manufactured under the reality of V_{th} and T variations. We use this metric to evaluate the power ranking taking into account both the average power and its variance caused by variation effects. Using this metric, we can see that the relative ranking for these lowest-power architectures still keeps the same trend as the ranking assuming no variation. From both the plots of our results, we observe that even though the average power increases when considering V_{th} and T effects, the relative ranking for the lowest-power network configurations doesn't vary much. This is because the ones that have lower power consumption also have typically simpler architectures using fewer network resources (thus more tolerant of the variations), and have longer latencies. Across our design space, we found out that typically torus plain and mesh plain are more tolerant to V_{th} and T variations than the architectures with express links or hierarchical links.

Even so, it is important to estimate power taking into account process and temperature variation because even though the lowest-power ranking may not vary much, a design may have a power envelope (constraint) within which the design is being optimized for performance. So, significant inaccuracies in total power dissipation may lead to incorrect design decisions as variations can introduce substantial differences in dissipation.

For the configurations that have lower power consumption, they are also the ones that have higher latency due to their relative simple router architectures and topology. So next we evaluate the EDPPF (energy delay product per flit) across our design space as this accounts for both performance and power dissipation. Figure 7 shows the first eight network configurations that have the lowest EDPPF with and without V_{th} and T variations considered. The x-axis in the figure lists, in increasing order of the EDPPF, these eight network architectures and configurations that have the lowest mode EDPPF when assuming no process variation with uniform temperature. The y-axis is the EDPPF normalized to the lowest EDPPF architecture under no variation, which is the torus plain (8,2). From the figure we can see that when assuming no variations, torus plain(8,2) has the lowest EDPPF. But when considering V_{th} and T variation effects, torus(4,2) actually has the lowest average EDPPF. Besides, torus (4,2) has lowest EDPPF variations across chips. So, designers might want to choose torus (4,2) when looking for configurations targeted at low EDPPF and if the layout/area constraints are feasible. From the figure, we can also notice that even though torus (4,1) has relatively higher average EDPPF among these top eight configurations, its EDPPF variance is much smaller than torus (16, 2) and torus (8, 4), so this factor should be taken into account when making design decisions. The upper part of the figure shows the associated product of mean and standard-deviation ($\sigma \times \mu$) metric. We use this metric to evaluate both the average EDPPF and the variance across chips together. Using this metric, we can see that torus plain(4,2) has lower $\sigma \times \mu$ than torus plain (8,2). Torus plain(8,1) now ranked third in EDPPF which is better than torus plain (4,4). An evident from the figure, upon considering the combined effects of mean and variance of EDPPF, the ranking becomes quite different when compared to the no variations scenarios. Designer taking into account several metrics and considering the effects of V_{th} and T variations, as we demonstrate, can significantly alter design decisions. Hence, the importance of considering effects of variations on power and performance during early design exploration.

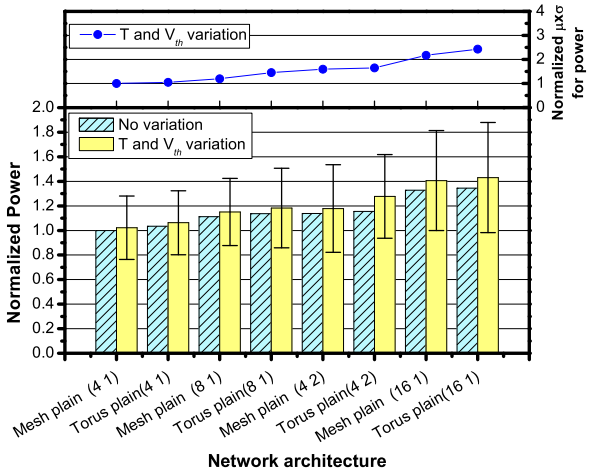

Figure 6. Normalized power considering T and V_{th} variation *topology_variant (buffer_size, number_of_virtual_channels)*

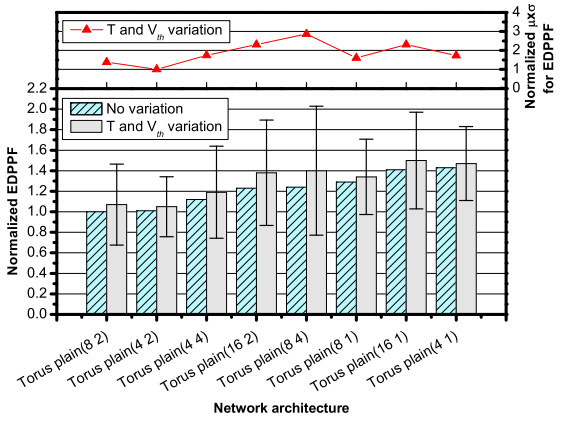

Figure 7. Normalized EDPPF considering T and V_{th} variation

4.3. Sensitivity analysis

Next we perform sensitivity analysis in the design space, i.e., the effect of the two considered variations are analyzed independently of each other.

4.3.1. Effect of temperature variation only

First, we study the impact of within-die temperature variation on power dissipation of interconnection networks, and compare the results against the experimental results that assume no process variation and a uniform temperature at 80 °C. Again, in this simulation, we assume that the temperature variation comes only from power dissipation variation across different interconnection network blocks on the chip (i.e., all the computation and storage blocks on the chip are held at the same, fixed power density for simplicity).

The x-axis of figure 8 presents the first eight network architectures or configurations that have the lowest power consumption when assuming no process variation and uniform temperature. The left and right-hand bars for each of these architectures shows power dissipation in presence of no variations and in presence of temperature variation only, respectively. The y-axis is the power consumption normalized to the lowest-power dissipating architecture, under no variation effects, which is mesh plain (4,1). From figure 8, we see that NoC power consumption increases from 2.2% to 6.4% with an average increase of 4% for these eight configurations, due to the impact of temperature variation across the chip. It is also seen that although the power consumption increases across all eight configurations upon considering temperature variation, the relative power ranking among them does not vary much for the reasons discussed before.

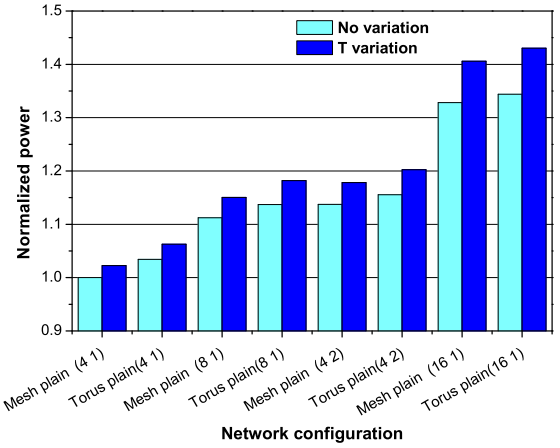

Figure 8. Power comparison

Next we evaluate the EDPPF among our design space. Figure 9 shows the first eight network configurations that have the lowest EDPPF with and without temperature variation consideration. From the figure we can see that when assuming uniform temperature across the chip, torus plain torus(8, 2) has the lowest EDPPF. But when considering temperature variation across the chip, torus plain torus(4, 2) actually has the lowest EDPPF. This is because temperature variation has more impact on power for torus(8, 2) than torus(4, 2). From the figure we can observe that the ranking is changed due to the effects of temperature variation. These results demonstrate that it is important to consider temperature variation at the early design stage as design decisions

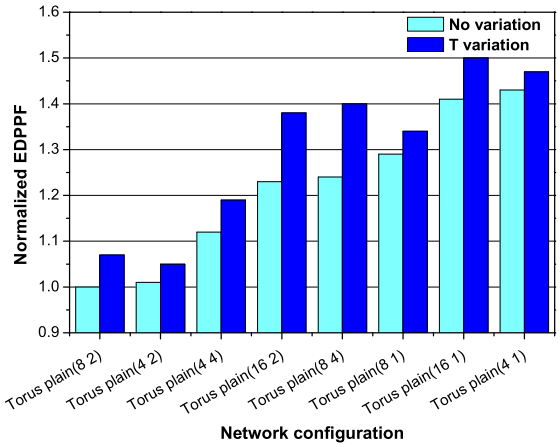

Figure 9. EDPPF comparison

can change.

In our simulation, we assume an initial temperature of 80 °C for each router region. Thermal simulation gives us a new temperature which in turn is used to compute new leakage power. This leakage power affects the power density which results in a new temperature estimation. This simulation "loop" typically lasts for 4 to 5 iterations before converging to stable temperature and leakage values. We also studied the dependence, on the initial temperature, of the number of iterations to converge and the final temperature reached. Our experiments show remarkable stability where neither the number of iterations nor the final converged value depend much on any reasonable starting temperature, as shown in figure 10.

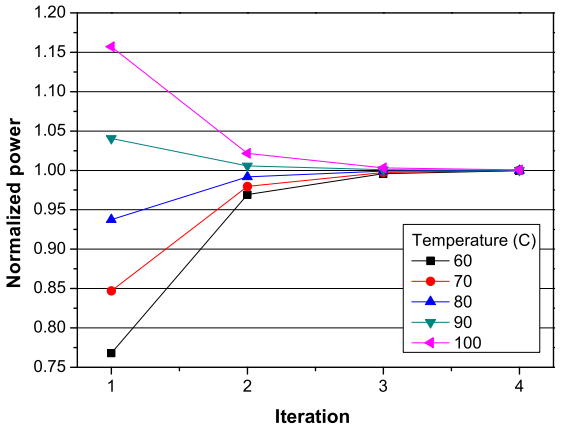

Figure 10. Thermal simulator initial temperature set on stability analysis

4.3.2. Effect of process variation only

When considering process variation for the power estimation, the power consumption is a distribution rather than a deterministic number as we explained before. Figure 11 shows that the power variance increases as the power con-

sumption rises. Since mesh plain and torus plain have relatively simpler router/topology and lower power consumption, the standard deviation of their power distributions are relatively small as well. From our experimental result, we found that typically torus plain and mesh plain are more tolerant to process variation than the architectures with express cubes or hierarchical links. In our simulation, torus with hierarchical link of interval 2 has the lowest latency, but their power variance is also much higher, making them less tolerant of process variations. This is because the network architectures which have higher performance are also the ones that use a lot of resources (larger buffer sets, more virtual channels), thus tend to be worse in tolerating process variations. (see figure 12). This figure shows that the 2D torus with hierarchical lik of interval 2 with 16 buffer-size and 4 virtual channels – plotted on the right side – has much higher power and spread/variance than the mesh-plain(4,1) plotted on the left. The latency for this mesh is, however, 1758 cycles as opposed to the 53-cycle latency of the said torus for a particular simulation of traffic.

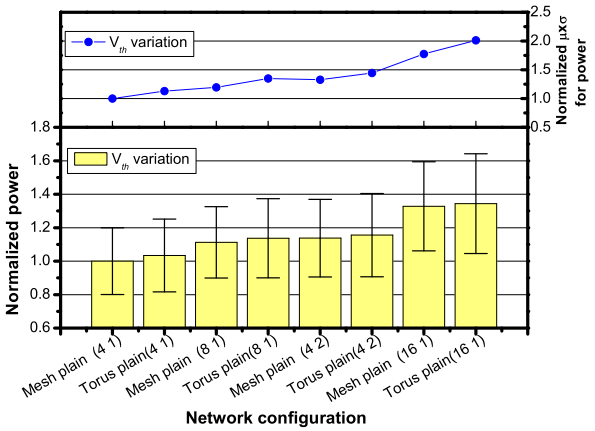

Figure 11. Normalized power vs. V_{th} variation

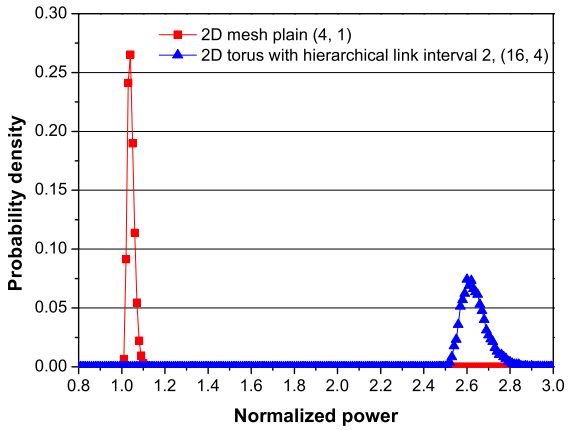

Figure 12. Normalized power for torus with hierarchical link interval 2 (16, 4) and mesh plain (4,1)

Figure 13 shows the EDPPF result when consider process variation. It shows that even though the standard deviation of EDPPF varies for different network architectures, the relative ranking among this first eigth network architectures do not vary much except for torus plain (4,1) now has lower $\sigma \times \mu$ than torus plain (16,1).

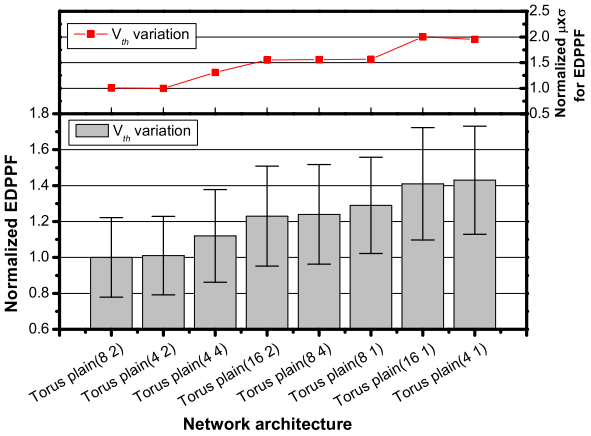

Figure 13. Normalized EDPPF for V_{th} variation

In summary, we found that it's important to consider and quickly and accurately estimate the impacts of both the process and temperature variations together. Otherwise, if considered separately, the design decisions can be significantly different from the reality where both factors occur simultaneously [36]. Furthermore, we showed the significance of considering both power and performance of designs as both those design properties are very important to practical designs. Although, we have focused on V_{th} for process variation, the other sources of variation such as L_{eff} can be incorporated in our work.

5. Conclusions

In this work, we implemented an architectural level power estimation simulator into a rapid design-space exploration toolchian (PolarisPT) which considers the effects of process and temperature variations for leakage-aware power estimation for NoCs. We then investigated how process and temperature variations affect the power consumption and energy-delay-product-per-flit for on-chip interconnection network designs. From the experiments, we can see that process and temperature influence the power consumption and EDPPF significantly enough to change designer's decision in choosing a network architecture. Not only is it quite important to take these factors into account, but also we should consider them simultaneously in the early design stages. We expect that as technology continues to scale, the importance of considering these effects will grow. For future work, we will extend it to study how process and temperature variations affect the latency and throughput of NoCs, along with the estimation's runtime efficiency.

References

[1] J. Tschanz, *et al.*, "Adaptive body bias for reducing impacts of die-to-die and within-die parameter variations on microprocessor frequency and leakage," *IEEE Journal of Solid-State Circuits*, vol. 37, no. 11, pp. 1396–1402, Nov 2002.

[2] A. Keshavarzi, *et al.*, "Effectiveness of reverse body bias for leakage control in scaled dual vt CMOS ICs," in *Proc. of the International Symposium on Low Power Electronics and Design*, 2001, pp. 207–212.

[3] P. Friedberg, *et al.*, "Modeling within-die spatial correlation effects for process-design co-optimization," in *Sixth International Symposium on Quality of Electronic Design ISQED2005*, March 2005, pp. 516–521.

[4] B. Stine, D. Boning, and J. Chung, "Analysis and decomposition of spatial variation in integrated circuit processes and devices," *IEEE Transactions on Semiconductor Manufacturing*, vol. 10, pp. 24–41, Feb. 1997.

[5] H. Chang and S. S. Sapatnekar, "Full-Chip analysis of leakage power under process variations, including spatial correlations," in *Proceedings. 42nd Design Automation Conference*, June 2005, pp. 523–528.

[6] A. Srivastava, *et al.*, "Accurate and effi cient gate-level parametric yield estimation considering correlated variations in leakage power and performance," in *ACM/IEEE Design Automation Conference*, June 2005, pp. 535–540.

[7] S. Zhang, V. Wason, and K. Banerjee, "A probabilistic framework to estimate full-chip subthreshold leakage power distribution considering within-die and die-to-die P-V-T variations," in *Proceedings of the 2004 International Symposium on Low Power Electronics and Design ISLPED'04*, 2004, pp. 156–161.

[8] R. R. Rao, *et al.*, "Parametric yield estimation considering leakage variability," in *ACM/IEEE Design Automation Conference*, June 2004, pp. 442–447.

[9] S. Mukhopadhyay and K. Roy, "Modeling and estimation of total leakage current in nano-scaled CMOS devices considering the effect of parameter variation," in *Proc. of International Symposium on Low Power Electronics and Design*, August 2003, pp. 172–175.

[10] S. Narendra, *et al.*, "Full-chip sub-threshold leakage power prediction model for sub-0.18 μm CMOS," in *Proc. of International Symposium on Low Power Electronics and Design*, August 2002, pp. 19–23.

[11] H. Su, *et al.*, "Full chip leakage estimation considering power supply and temperature variations," in *Proceedings of the 2003 International Symposium on Low Power Electronics and DesignISLPED'03*, August 2003, pp. 78–83.

[12] E.Humenay, "Toward an architectural treatment of parameter variations," Univ. of Virginia dept. of computer science, Tech. Rep. CS-2005, 2005.

[13] C. Saumya, *et al.*, "Considering process variations during system-Level power analysis," in *Proc. of the International Symposium on Low Power Electronics and Design*, Oct 2006, pp. 342–345.

[14] E. Humenay, D. Tarjan, and K. Skadron, "Impact of parameter variations on multi-core chips," in *Proc. Wkshp. on Architecture Support for Gigascale Integration*, 2006.

[15] D. Marculescu and E. Talpes, "Energy awareness and uncertainty in microarchitecture-level design," *IEEE Micro*, vol. 25, no. 5, pp. 64–76, Sept.-Oct. 2005.

[16] N. Azizi, *et al.*, "Variations-aware low-power design with voltage scaling," in *Proceedings. 42nd Design Automation Conference*, June 2005, pp. 529–534.

[17] W. J. Dally and B. Towles, "Route packets not wires: On-chip interconnection networks," in *Proc. Design Automation Conf.*, June 2001.

[18] L. Benini and G. De Micheli, "Networks on chips: A new SoC paradigm," *IEEE Computer*, vol. 35, no. 1, pp. 70–78, Jan. 2002.

[19] J. Kim, *et al.*, "Energy characterization of a tiled architecture processor with on-chip networks," in *Proc. of the 8^{th} International Symposium on Low Power Electronics and Design (ISLPED'03)*, August 2003, pp. 424–427.

[20] S. Vangal, *et al.*, "A 5.1ghz 0.34mm^2 Router for Network-on-Chip Applications," in *2007 IEEE Symposium on VLSI Circuits*, 2007.

[21] V. Soteriou, *et al.*, "Polaris: A System-Level roadmapping toolchain for On-Chip interconnection networks," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 15, no. 8, pp. 855–868, Aug 2007.

[22] S. P. Mohanty, *et al.*, *Low-Power High-Level Synthesis for Nanoscale CMOS Circuits*, 1st ed. Springer, ISBN:978-0387764733, April 2008.

[23] "Predictive technology model," www.eas.asu.edu/ ptm.

[24] Intel, personal communication.

[25] "Polaris," http://www.princeton.edu/ eisley/polaris.html.

[26] V. Soteriou, H. Wang, and L. Peh, "A statistical traffi c model for on-Chip interconnection networks," in *14th IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS)*, Monterey, California, September 2006, pp. 104–116.

[27] N. Eisley and L.-S. Peh, "High-level analysis for on-chip networks," in *Proc. of the 7^{th} International Conference on Compilers, Architecture and Synthesis for Embedded Systems (CASES)*, 2004.

[28] H.-S. Wang, *et al.*, "Orion: A power-performance simulator for interconnection networks," in *Proceedings. 35th Annual IEEE/ACM International Symposium on Microarchitecture*, Nov. 2002, pp. 294–305.

[29] Semiconductor Industry Association, "International technology roadmap for semiconductors."

[30] X.-N. Chen and L.-S. Peh, "Leakage power modeling and optimization of interconnection networks," in *Proceedings of the 2003 International Symposium on Low Power Electronics and DesignISLPED'03*, Aug. 2003, pp. 90–95.

[31] Y. Zhang, *et al.*, "Hotleakage: A temperature-aware model of subthreshold and gate leakage for architects," University of Virginia, Tech. Rep. CS-2003-05, March 2003.

[32] U. C. Berkeley., "BSIM3v3.1 SPICE MOS device models," http://www-device.EECS.Berkeley.EDU/bsim3/, 1997.

[33] J. Butts and G. Sohi, "A static power model for architects," in *Proceedings. 33rd Annual IEEE/ACM International Symposium on Microarchitecture MICRO-33*, 2000, pp. 191–201.

[34] Y. Yang, *et al.*, "Adaptive chip-package thermal analysis for synthesis and design," in *Proc. of IEEE Conf. on Design, Automation, and Test in Europe (DATE'06)*, March 2006, pp. 1–6.

[35] S. Vangal, *et al.*, "An 80-Tile 1.28TFLOPS Network-on-Chip in 65*nm* CMOS," in *IEEE International Solid-State Circuits Conference ISSCC2007*, Feb. 2007, pp. 98–589.

[36] S. P. Mohanty, *et al.*, "Interdependency Study of Process and Design Parameter Scaling for Power Optimization of Nano-CMOS Circuits under Process Variation," in *Proceedings of the 16th ACM/IEEE International Workshop on Logic and Synthesis (IWLS)*, 2007, pp. 207–213.