Performance Evaluation of Congestion-Aware Routing with DVFS on a Millimeter-Wave Small World Wireless NoC

JACOB MURRAY, Washington State University
RYAN KIM, Washington State University
PAUL WETTIN, Washington State University
PARTHA PRATIM PANDE, Washington State University
BEHROOZ SHIRAZI, Washington State University

The mm-wave small-world wireless NoC (mSWNoC) has emerged as an enabling interconnection infrastructure to design high-bandwidth and energy-efficient multicore chips. In this mSWNoC architecture, long-range communication predominately takes place through the wireless shortcuts operating in the range of 10-100 GHz, whereas the short-range data exchange occurs through conventional metal wires. This results in performance advantages (lower latency and energy dissipation) mainly stemming from using the wireless links as long-range shortcuts between far apart cores. The performance gain introduced by the wireless channels can be enhanced further if the wireline links of the mSWNoC are optimized according to the traffic patterns arising out of the application workloads. While there is significant energy savings, and hence temperature reduction, in the network due to the mSWNoC architecture, a load-imbalanced network is still susceptible to local temperature hotspots. In this work, we demonstrate that by incorporating congestion-avoidance routing with network-level dynamic voltage and frequency scaling (DVFS) in an mSWNoC, the power and thermal profiles can be improved without a significant impact on the overall network performance. In this work we demonstrate how novel interconnect architectures enabled by the on-chip wireless links coupled with power management strategies can improve the energy and thermal characteristics of a mSWNoC significantly without introducing any performance degradation with respect to the conventional mesh-based NoC.

Categories and Subject Descriptors: C.2.1 [Computer-Communication Networks]: Network Architecture and Design

General Terms: Design, Algorithms, Performance

Additional Key Words and Phrases: Multi-core, NoC, Small-World, Wireless Links, DTM, DVFS

1. INTRODUCTION

The existing method of implementing a NoC with planar metal interconnects is deficient due to high latency, significant power consumption, and temperature hotspots arising out of long, multi-hop wireline paths used in data exchange. It is possible to design high-performance, robust, and energy-efficient multicore chips by adopting novel architectures inspired by complex network theory in conjunction with on-chip wireless links. Networks with the small-world property have very short average path lengths, making them particularly interesting for efficient communication with minimal resources. Using the small-world approach we can build a highly efficient NoC with both wired and wireless links. Neighboring cores should be connected through traditional metal wires while widely separated cores will communicate through long-range, single-hop, wireless links.

A small-world network principally has an irregular topology [Petermann et al. 2005]. Routing in irregular networks is more complex, because routing methods need to be topology agnostic. Hence, it is necessary to investigate suitable routing mechanisms.
for small-world networks. Routing in irregular networks can be classified into two broad categories, viz., rule- and path-driven strategies [Flich et al. 2012]. Rule-driven routing is typically done by employing a spanning tree for the network. Messages are routed along this spanning tree with specific restrictions to achieve deadlock freedom. Because deadlock freedom is taken into account first for these routing strategies, minimal paths through the network for every source-destination pair cannot be guaranteed [Flich et al. 2012]. Conversely, for path-driven routing, minimal paths between all source-destination pairs are first guaranteed and then deadlock freedom is achieved by preventing portions of traffic from using specific resources, such as the virtual channels [Flich et al. 2012]. These routing strategies are not only necessary to achieve deadlock freedom for performance gains; they also influence the power dissipation of the NoC.

The power and thermal profiles of the network depend on how efficiently the routing mechanism can move the traffic through the network and balance the traffic within the network elements. For irregular network architectures like the wireless NoC considered here, topology-agnostic efficient routing schemes facilitate implementation of Dynamic Thermal Management (DTM) in order to avoid high spatial and temporal temperature variations among NoC switches and links, thereby mitigating local network hotspots. Recent works on DTM [Chaparro et al. 2007] for multicore architectures focus on optimizing the performance of the cores only and explores the design space in the presence of thermal constraints. However, the performance of a multicore chip is also heavily influenced by its overall communication infrastructure, which is predominantly a NoC. Depending on the application, the network can consume a significant portion (around 50% or more) of the chip’s overall power budget [Murray et al. 2013]. Hence, power and thermal management in the network level is important. To achieve DTM, the above-mentioned irregular network routing strategies can be enhanced to incorporate congestion avoidance, which attempts to mitigate the relative network-level hotspots of mSWNoC. Congestion avoidance can be achieved by either congestion prevention or congestion detection/recovery. We demonstrate that by incorporating appropriate congestion-aware techniques on top of the irregular network routing strategies, it is possible to reduce local temperature hotspots in mSWNoCs without a significant performance impact. Furthermore, we propose incorporating suitable Dynamic Voltage and Frequency Scaling (DVFS) techniques jointly with the irregular routing strategies to further reduce temperatures of the NoC.

The proposed methodologies are general, in the sense that we are not bound to the specific network routing and DVFS mechanisms that we consider. Instead, we focus on demonstrating how to enhance the power and thermal profiles of mSWNoC-enabled multicore chips by complementing the advantages introduced by the interconnect architecture. In this paper we study the importance of suitable DVFS mechanisms that exploit both the architecture and the routing strategies to improve the overall power and thermal profiles. We consider network latency, energy dissipation, and the network switch and link temperatures as the relevant metrics in the performance evaluation. We demonstrate that depending on the specific benchmarks, coupled with the DTM and DVFS methodologies explained in this work, the power-thermal-performance (PTP) trade-offs could vary greatly. The capabilities of the network in conjunction with these power and thermal management strategies drastically reduce energy consumption of the NoC.
2. RELATED WORK

The limitations and design challenges associated with existing NoC architectures are elaborated in [Marculescu et al. 2009]. Conventional NoCs use multi-hop, packet-switched communication. At each hop, the data goes through a complex router/switch, which contributes considerable power, throughput, and latency overheads. To improve performance in a regular mesh, a methodology to automatically synthesize architectures with a few application specific long-range links was proposed in [Ogras et al. 2005]. Subsequently, performance advantages of NoCs by insertion of long-range wireline links following the principles of small-world graphs were elaborated in [Ogras et al. 2006]. The concept of express virtual channels is introduced in [Kumar et al. 2008]. Despite significant performance gains in the above schemes, the long-range links are designed with conventional wires. It is already shown that beyond a certain length, wireless links are more energy-efficient than conventional metal wires. Hence, the performance improvements by using long-range wireless links will be more than that using wireline links [Deb et al. 2012b].

A comprehensive survey regarding various Wireless NoC (WiNoC) architectures and their design principles are presented in [Deb et al. 2012a]. WiNoC architectures can be divided into two sub categories, viz. mesh with wireless links inserted on top of it, and hierarchical architectures with long-range wireless shortcuts. Among the first category, notable examples include design of a WiNoC based on CMOS ultra wideband (UWB) [Zhao et al. 2008], 2D concentrated mesh-based WCube architecture using sub-THz wireless links [Lee et al. 2009], and the inter-router wireless scalable express channel for NoC (GWISE) architecture [Di Tomaso et al. 2011]. Possibilities of creating novel architectures aided by the on-chip wireless communication have been explored in [Ganguly et al. 2011a] and [Chang et al. 2012]. These two works proposed design of hierarchical and hybrid WiNoC architectures using long-range wireless shortcuts. The whole system is partitioned into multiple small clusters of neighboring switches called subnets. In the upper level of the network, the subnets are connected via wireline and wireless links. In both these designs, the subnets are connected in a regular structure like a mesh or a ring, in the second level of the hierarchy, long-range wireless shortcuts are placed on top of that. It is also shown that a WiNoC, where the network architecture is designed following the power-law based small-world connectivity [Ganguly et al. 2011b], is more robust in presence of wireless link failures compared to the hierarchical counterpart [Ganguly et al. 2011b].

DVFS is a popular methodology to optimize the power usage/heat dissipation of electronic systems without significantly compromising overall system performance [Garg et al. 2012]. Many of the existing works principally address power and thermal management strategies for the processing cores only. The network consumes a significant part of the chip’s power budget; greatly affecting overall temperature. Consequently, power and thermal management at the network-level is important. ThermalHerd, proposed in [Shang et al. 2006], provides a distributed runtime scheme for thermal management that allows NoC routers to collaboratively regulate the network temperature profile and work to avert thermal emergencies while minimizing performance impact. A Voltage Frequency Island (VFI) power management technique for processing elements and routers was proposed in [Bogdan et al. 2013]. On the Intel Single-chip Cloud Computer (SCC), a DVFS scheme was implemented in [David et al. 2012]. By tuning the frequencies of network routers depending on traffic patterns, the NoC power profile can be improved [Mishra et al. 2009]. There has also been significant work for congestion-aware routing for regular NoC architectures [Ebrahimi et al. 2012a; Ebrahimi et al. 2012b; Kia et al. 2011]. All
these works primarily target mesh-based wireline NoCs. It is already shown that WiNoC improves the temperature profile of the NoC switches and links compared to a traditional mesh in presence of DVFS [Wettin et al. 2013]. However, for WiNoCs, hotspots are still localized to specific network switches that handle high traffic loads.

In [Sabry et al. 2014], a survey for thermal management of 3D MPSoCs was performed. The concept of emergence was explored in [Hollis et al. 2014]. In this work, the authors proposed a reconfigurable topology to improve performance and reduce energy consumption. While most of these works rely on a regular mesh structure as the backbone for the NoC, we propose network-level power and thermal management for the WiNoC architecture under consideration by incorporating topology-agnostic routing. Our main aim is to evaluate the performance of topology-agnostic and congestion-aware routing-based DTM followed by DVFS to avoid creation of network temperature hotspots in WiNoCs.

3. WIRELESS NOC ARCHITECTURE AND COMMUNICATION MECHANISM

Small-world graphs are characterized by many short-distance links between neighboring nodes as well as a few relatively long-distance direct shortcuts. In this work, we consider a mm-wave small-world wireless NoC architecture (mSWNoC), where the long-range shortcuts are implemented through mm-wave wireless links operating in 10-100 GHz range. Fig. 1 represents such an mSWNoC with 16 cores, where each core is associated with a NoC switch (not shown for clarity). The mSWNoC has many short-range local links, as well as, a few long-range shortcuts schematically represented by the arching, dashed, interconnects. As an example, it also has three nodes with transceivers to create wireless channels. In the following sections we discuss the characteristics of this mSWNoC architecture and analyze its performance and temperature profiles in the presence of congestion-aware adaptive routing and DVFS techniques.

3.1 Topology of mSWNoC

In the mSWNoC topology, each core is connected to a switch and the switches are interconnected using both wireline and wireless links. The topology of the mSWNoC is a small-world network where the links between switches are established following a power-law model [Wettin et al. 2013], [Watts et al. 1998]. In this small-world network there are still several long wireline interconnects. As these are extremely costly in terms of power and delay, we use mm-wave wireless links to connect switches that are separated by a long distance. In [Deb et al. 2012c], it is demonstrated that it is possible to create three non-overlapping channels with

![Figure 1. mSWNoC architecture with short- and long-range links.](image)
central frequencies of 31 GHz, 57.5 GHz, and 120 GHz respectively. Using these three channels we overlay the wireline small-world connectivity with the wireless links such that a few switches get an additional wireless port. Each of these wireless ports will have wireless interfaces (WIs) tuned to one of three different frequency channels. More frequently communicating WIs are assigned to the same channel to optimize the overall hop-count. One WI is replaced by a gateway WI that has all three channels assigned to it; this facilitates data exchange between the non-overlapping wireless channels. To have a detailed comparative performance evaluation of the mSWNoC we also consider a wireline-only small-world NoC (SWNoC) topology. The SWNoC topology is designed identically as the mSWNoC. However, the SWNoC has no wireless links. The long-range shortcuts are implemented through multi-hop metal wires.

We have assumed an average number of connections from each switch to the other switches, \(k\). The value of \(k\) is chosen to be 4 so that the mSWNoC does not introduce any additional switch overhead with respect to a conventional mesh. Also an upper bound, \(k_{\text{max}}\), is imposed on the number of wireline links attached to a particular switch so that no switch becomes unrealistically large in the mSWNoC. This also reduces the skew in the distribution of links among the switches. Both \(k\) and \(k_{\text{max}}\) do not include the local NoC switch port to the core. For a 64-core system the optimum value of \(k_{\text{max}}\) was found to be 7 [Wettin et al. 2013].

### 3.2 Wireless Interface

The two principal WI components are the antenna and the transceiver. The on-chip antenna for the mSWNoC has to provide the best power gain for the smallest area overhead. A metal zigzag antenna has been demonstrated to possess these characteristics [Deb et al. 2012b]. This antenna also has negligible effect of rotation (relative angle between transmitting and receiving antennas) on received signal strength, making it most suitable for mm-wave NoC applications. Zigzag antenna characteristics depend on physical parameters like axial length, trace width, arm length, bend angle, etc. By varying these parameters, the antennas are designed to operate on different frequency channels [Deb et al. 2012b]. To ensure high throughput and energy efficiency, the WI transceiver circuitry has to provide a very wide bandwidth, as well as low power consumption. Non-coherent on-off keying modulation is chosen, as it allows relatively simple and low-power circuit implementation. The transmitter consists of an up-conversion mixer and a power amplifier. In the receiver, a direct-conversion topology is used, consisting of a low noise amplifier, a down-conversion mixer and a baseband amplifier. An injection-lock voltage-controlled oscillator is reused for both the TX and RX. The detailed description of the transceiver circuitry is out of the scope of this paper. However, the transceiver was designed following [Deb et al. 2012b]. With a data rate of 16 Gbps, the wireless link dissipates 1.95 pJ/bit. The total area overhead per wireless transceiver is 0.25mm\(^2\).

### 3.3 Flow Control

In the mSWNoC, data is transferred via a flit-based, wormhole routing [Pande et al. 2005]. Between a source and destination pair, the wireless links, through the WIs, are only chosen if the wireless path reduces the total path length compared to the wireline path. This can potentially give rise to hotspot situations in the WIs. Many messages will try to access the WI shortcuts simultaneously, thus overloading the WIs, which would result in higher latency and energy dissipation. Token flow control [Kumar et al. 2008b] and distributed routing are used to alleviate this problem. An arbitration mechanism is designed to grant access to the wireless medium to a particular WI, including the gateway WI, at a given instant to avoid interference and
contention between the WIs that have the same frequency. To avoid centralized control and synchronization, the arbitration policy adopted is a wireless token passing protocol [Chang et al. 2012]. In this scheme, a single flit circulates as a token in each frequency channel. The particular WIs possessing the wireless tokens can broadcast flits into the wireless medium in their respective frequencies. The wireless token is forwarded to the next WI operating in the same frequency channel after all flits belonging to a message at the WI are transmitted. Packets are rerouted, through wireline links, if the WI buffers are full and it does not have the token. In case the WI’s buffers are full, or if the WI does not have the token, packets attempting to access the WI need to be rerouted. As rerouting packets can potentially lead to deadlock, a rerouting strategy similar to Dynamic Quick Reconfiguration (DQR), as presented in [Sem-Jacobsen et al. 2012], is used to ensure deadlock freedom. In this situation, the current WI becomes the new source for the packet, which is forced to take a wireline only path to the final destination, while following the original routing restrictions that are detailed in section 4 impose.

4. ROUTING AND DYNAMIC THERMAL MANAGEMENT (DTM)

The power law connectivity-based mSWNoC principally has an irregular network topology. Routing in irregular networks is more complex, because the routing methods need to be topology agnostic. Hence, it is necessary to investigate suitable routing mechanisms for mSWNoCs to ensure deadlock-free routing. As mentioned earlier, irregular networks can have routing strategies that are either rule- or path-driven [Flich et al. 2012]. For comparison, we have adopted routing strategies from both categories as earlier described.

4.1 ALASH-based Routing

The first routing strategy is an adaptive layered shortest-path routing (ALASH) algorithm [Flich et al. 2012], which belongs to the path-based classification. ALASH is built upon the layered shortest path (LASH) routing algorithm, but has more flexibility by allowing each message to adaptively switch paths, letting the message choose its own route at every intermediate switch. Here, ALASH inherently has congestion avoidance built in, and employs the congestion prevention approach.

The LASH algorithm takes advantage of the multiple virtual channels in each switch port of the NoC switches in order to route messages along the shortest physical paths. In order to achieve freedom from deadlock, the network is divided into a set of virtual layers, which are created by dedicating the virtual channels from each switch port into these layers. The shortest physical path between each source-destination pair is then assigned to a layer such that the layer’s channel dependency graph remains free from cycles. A channel dependency is created between two links in the source-destination path when a link from switch $i$ to switch $j$ and a link from switch $j$ to switch $k$ satisfies the following condition,

$$\text{pathlength}(i) < \text{pathlength}(j) < \text{pathlength}(k)$$  \hspace{1cm} (1)

where $\text{pathlength}(X)$ is the length of the minimal path between switch $X$ and the original source switch. When a layer’s channel dependency graph has no cycles, it is free from deadlocks as elaborated in [Flich et al. 2012].

In order to increase the adaptability of the routing, multiple shortest paths between all source-destination pairs are found and then included into as many layers as possible. The message route through the network depends on the layers each source-destination pair use. Therefore, the layering function that controls how the layers are allocated for each source-destination pair has an impact on the latency, energy, and
thermal profile of the mSWNoC. The layering function that we use in this work uses apriori knowledge of the frequency of traffic interactions, $f_{ij}$, and allocates as many layers as possible to source-destination pairs with high $f_{ij}$. This improves the adaptability of messages with higher $f_{ij}$ by providing them with greater routing flexibility. It is possible to induce deadlock if a message is allowed to continuously switch back and forth between two or more layers. Hence, a message is not allowed to revisit a layer that it has previously traveled in to maintain deadlock freedom.

For ALASH, the decision to switch paths is based on current network conditions such as virtual channel availability and current communication density of the network. For example, in [Qian et al. 2012], the channel fitness of the links was determined using the average waiting time and channel availability. Similarly, in this work, to estimate the congestion of a particular part of the network we utilize the average communication density as the relevant parameter, which is defined by (2) below,

$$\rho_{\text{comm}_i} = \frac{N_i}{W_s}$$

where, $\rho_{\text{comm}}$ is the communication density for switch or link $i$ expressed in flits per cycle, $W_s$ is the monitoring window size in cycles, and $N_i$ is the number of flits handled by switch or link $i$ in the monitoring window. The higher $\rho_{\text{comm}}$ a switch has the more power it dissipates in the given window. When the switch energy increases, the temperature increases. Hence, the $\rho_{\text{comm}}$ of the switch can be used to monitor its thermal behavior. Therefore, at every switch in the path, ALASH makes an evaluation regarding which link to follow based on $\rho_{\text{comm}}$ in order to reduce congestion in the network. This is done by choosing the path $j$ that minimizes $\rho_{\text{path}}$ given in (3) below,

$$\rho_{\text{path}} = \sum_{i=1}^{\text{pathlength}(j)} \rho_{\text{comm}_{\text{path}(j)}}$$

where, $\text{path}(j,i)$ represents the $i^{th}$ link of the $j^{th}$ path, $\rho_{\text{comm}}$ is the communication density defined in (2) and $\text{pathlength}(j)$ is the number of links in the path $j$. Since ALASH always takes the shortest physical path, this type of adaptive routing does not add extra hops to the path. As $\rho_{\text{comm}}$ is used to determine the suitable paths for the data packets, this helps to evenly distribute the traffic and reduce congestion in the network. By balancing the load across the network, this type of routing is inherently beneficial to improve the thermal profile of the mSWNoC.

### 4.2 MROOTS-based Routing

To ensure deadlock-free routing the second routing strategy we consider, belonging to the rule-based classification, is an up/down tree-based routing methodology that utilizes a multiple tree roots (MROOTS)-based mechanism [Lynse et al. 2006]. By allowing multiple routing trees to exist, where each tree routes on a dedicated virtual channel, traffic bottlenecks in the upper tree levels that is inherent in this type of routing can be reduced. We adopt a traffic-weighted, minimized hop-count, root-node placement policy. Selecting $M$ tree roots will create $M$ trees in the network, where the chosen $M$ roots minimize the optimization metric $\mu$ as defined in (4) below.

$$\mu = \min_{\text{roots}} \left( \sum_{i} \sum_{j} h_{ij} f_{ij} \right)$$

Here, the minimum path distance in hops, $h_{ij}$, from switch $i$ to switch $j$ is determined following the up/down routing strategy [Lynse et al. 2006]. The frequency of traffic interaction between the switches is denoted by $f_{ij}$. As root selection only affects valid routing paths for deadlock freedom and does not alter the physical placement of links, any apriori knowledge of the frequency of traffic interaction aids in root selection. Incorporating $f_{ij}$ helps minimize the routed path lengths for specific workloads on the mSWNoC architecture. All wireless and wireline links that are not
a part of the tree are reintroduced as shortcuts, which are considered up paths in both directions. An allowed route never uses an up direction along the tree after it has been in the down path once. In addition, a packet traveling in the downward direction is not allowed to take a shortcut, even if that minimizes the distance to the destination. Hence, channel dependency cycles are prohibited, and deadlock freedom is achieved [Lynse et al. 2006].

A congestion-aware adaptive routing strategy was built upon MROOTS (AMROOTS) following the congestion detection and recovery methodology. As MROOTS does not guarantee shortest path, using the ALASH-based congestion-aware technique would only produce longer routing paths. Hence, we implement the congestion detection and recovery method on MROOTS. AMROOTS was developed to 1) monitor and 2) avoid forming local hotspots. This technique was developed following an approach similar to [Shang et al. 2006].

First, a threshold, $\beta$, was set in order to detect which switches have relatively high $\rho_{comm}$ as described in equation (2) above. If the $\rho_{comm}$ of a switch exceeds $\beta$ by the end of a given monitoring window, that switch is put on an avoidance list. If a switch’s $\rho_{comm}$ continuously exceeds $\beta$, it is avoided exponentially to the number of times $\beta$ is exceeded. When a message is generated in the network, the path with the smallest distance to the destination should be taken to minimize latency, and maximize throughput. However, if that path contains switches on the avoidance list, that path should only be taken with a certain probability. By attempting to avoid a path with a hotspot switch, we minimize the amount of traffic that the hotspot switch may handle, however we do not completely block the path to all traffic. The probability of following any path is determined by the path ratio, $R_{path}$, given in (5),

$$R_{path} = \begin{cases} 1, & \text{unthrottled path} \\ \frac{1}{L} \prod_{i=1}^{L} r_i, & \text{otherwise} \end{cases}$$

(5)

where, an unthrottled path (a path with no switches on the avoidance list) has a path ratio of 1, and a throttled path has a path ratio defined above in (5). Here, $L$ is the path length, and $r$ is $i$th switch’s throttling ratio, defined in (6).

$$r_i = \left(\frac{1}{NT_i + 1}\right)^{NT_i}$$

(6)

In (6), $NT_i$ is the number of consecutive throttle attempts for a given switch (the number of times the switch has been on the avoidance list consecutively). As can be seen in (6), the longer a switch remains on the avoidance list, it is avoided with larger probability. This throttling methodology was developed following [Shang et al. 2006], in which a similar exponential weight is given to determine the appropriate path selection. For a particular source-destination pair, a list of ranked paths is considered. The $n^{th}$ ranked path ratio, $R_n$, is determined starting with the lowest ranked path and continuing until an unthrottled path is found, or all of the paths have been checked. The ranked paths are then compared probabilistically in order to determine which path will be selected for routing.

For the mesh architecture, we develop a congestion-aware routing scheme similar to [Shang et al. 2003b] where data belonging to all except one virtual channel, follow a regular X-Y routing strategy. If a switch exceeds the threshold $\beta$, it uses the hotspot avoidance methodology described earlier in this subsection. When a packet is injected into the network, if its X-Y route would pass through a switch on the avoidance list, it is placed on the last virtual channel that is used specifically for rerouting.
5. JOINT DTM/DVFS

The small-world architectures and the congestion-aware routing schemes discussed in section 4, modify the distribution of network traffic patterns significantly. Hence, it is possible to tune the voltage and frequency of the small-world switches and links depending on the traffic-dependent bandwidth requirements.

First, by reducing the hop count between largely separated communicating cores, wireless shortcuts have been shown to carry a significant amount of the overall traffic within an mSWNoC [Wettin et al. 2013]. The amount of traffic detoured in this way is substantial and the low-power wireless links enable energy savings. However, the energy dissipation within the network is still dominated by the data traversing the wireline links. Hence, the overall energy dissipation of the mSWNoC can be improved even further if the characteristics of the wireline links are optimized. Consequently, implementing DVFS on the wireline links and the associated switch ports of an mSWNoC-enabled multicore architecture has the potential for more energy savings [Wettin et al. 2013]. Secondly, as discussed in section 4, ALASH and AMROOTS inherently reduce traffic relayed through relative hotspot switches. Attempting to evenly distribute traffic among the network elements to reduce congestion does this for ALASH. In AMROOTS, by explicitly routing away from relative hotspot switches, the amount of traffic on these switches will be significantly reduced. As traffic that is not created nor destined for the given hotspot switch will avoid it, if possible, the utilization of links associated with the hotspot switch will be lowered, provided that the switch remains on the avoidance list. Hence, in this work, we propose two types of DVFS schemes where switches tune the voltages and frequencies of their corresponding ports and associated links depending on the nature of the traffic.

For ALASH, a DVFS strategy is used that combines both past (reactive) and future, (proactive) knowledge of the link utilizations to decide the voltage and frequency of the switch ports and associated links. For AMROOTS, a reactive DVFS strategy is first employed on non-hotspot switches, where the ports and links are tuned according to their utilizations. Then, a proactive DVFS strategy is employed where knowledge of the network, such as hotspot switches and communication densities are used to estimate its impact on other sections of the network and tune the ports and links accordingly. Hence, every switch can utilize proactive and/or reactive DVFS given its traffic-dependent, relative hotspot characteristics.

5.1 Reactive DVFS

A method for history based DVFS was proposed in [Shang et al. 2003a] and adopted for mSWNoC in [Wettin et al. 2013]. This type of DVFS is used as the base DVFS algorithm for both the ALASH- and AMROOTS-based mechanisms. In this scheme, every NoC switch predicts future traffic patterns based on an exponential weighted average history determined for each link. The short term link utilization is characterized by,

\[ U_{\text{short}} = \frac{1}{H} \sum_{i=1}^{H} b_i \]  \hspace{1cm} (7)

where, \( H \) is the prediction window, and \( b_i \) is 1 if a flit traversed the link on the \( i^{th} \) cycle of the history window, and 0 otherwise. The predicted future link utilization, \( U_{\text{Predicted}} \), is an exponential weighted average determined for each link according to,

\[ U_{\text{Predicted}} = \frac{W \cdot U_{\text{short}} + U_{\text{Predicted}(t-1)}}{W+1} \] \hspace{1cm} (8)

where, \( U_{\text{Predicted}(t-1)} \) is \( U_{\text{Predicted}} \) for the previous DVFS window and \( W \) is the weight
between the current history window and the previous windows. As mentioned above we propose to combine this history-based prediction with the ALASH and AMROOTS routing decisions respectively to implement the overall DVFS mechanism.

5.2 Proactive DVFS

As the basic path selection method depends on the exact routing methodology, the exact proactive DVFS implementation varies between ALASH and AMROOTS.

In ALASH, at any particular time, every switch knows which link a message is going to take to reach a particular destination. This information can be used to predict the future link utilizations for the links connected to the switch. If there aren’t many messages destined for a particular link according to the ALASH decision-making mechanism, then the utilization for that link in the near-future can be predicted to be low. This prediction, $U_{\text{Proactive}}$, is defined below

$$U_{\text{Proactive}} = \sum_{i=0}^{N-1} [D(j) \cdot \sum_{i=1}^{H} B_i(j)]$$

where, $N$ is the number of switches in the network, $B_i(j)$ is 1 if a flit destined for switch $j$ traversed the link on the $i^{th}$ cycle of the history window and 0 otherwise, and $D(j)$ is 1 if the next flit destined for switch $j$ will use the link and 0 otherwise. $U_{\text{Proactive}}$ from (9) is used with $U_{\text{predicted}}$ from (8), to create a new prediction, $U_{\text{ALASH}}$, defined as,

$$U_{\text{ALASH}} = \sigma \cdot U_{\text{predicted}} + \gamma \cdot U_{\text{Proactive}}$$

(10)

where, $\sigma$ and $\gamma$ are weight parameters, for $U_{\text{predicted}}$ and $U_{\text{Proactive}}$ respectively, that are optimized for a certain latency performance requirement point which will be described in section 6.2. Using (10), we are able to control how much DVFS relies on the history, using weight $\sigma$, and on the future routing, using weight $\gamma$.

As the AMROOTS routing strategy is implemented on the mSWNoC architecture, there is some inherent knowledge in the avoidance list, which can be useful for tuning the voltages and frequencies of the various switch ports and associated links. When a switch is added to the avoidance list, utilization will be heavily reduced, as traffic will try to avoid routing through the hotspot. Hence, a method for performing proactive DVFS was adopted. The proactive DVFS tunes the voltage/frequency (V/F) of a given hotspot switches’ ports and associated links down when the switch is on the avoidance list. As our first priority is to reduce switch temperature hotspots as much as possible, proactive DVFS combined with the adaptive routing strategy is important to help reduce the affected hotspot quickly. This is done by not only mitigating much of the through-switch traffic, but by also reducing the V/F of the affected switches’ ports and links to allow it to cool down faster. This DVFS methodology is also implemented on the mesh architecture for undertaking detailed comparative evaluation.

After $H$ cycles have elapsed, where $1/H$ is the maximum allowable V/F switching rate, the NoC switch determines whether a given link’s predicted utilization meets a specific threshold. By allowing thresholds at several different levels of predicted utilizations, a finer-grain balance between energy savings, due to lowering the voltage and frequency, and latency penalty from mispredictions, can be obtained. After $H$ cycles, the algorithm determines if DVFS should be performed on the switch ports and associated links based on the predicted bandwidth requirements of future traffic. Depending on which threshold was crossed, if any, the switch then determines whether or not to tune the V/F of the ports and associated links. Fig. 2 shows the flow charts of the proactive/reactive DVFS methodologies for both ALASH and AMROOTS. In order to prevent a direct multi-threshold jump, which would cause
high delay and energy overhead, the voltage and frequency can step up or down once, or remain unchanged during one V/F transition. After each adjustment of the V/F on a given link, energy savings and latency penalty was determined.

Voltage regulators are required to step up or step down voltage in order to dynamically adjust voltage, and hence frequency. By using on-chip voltage regulators with fast transitions, latency penalties and energy overheads due to voltage transitions can be kept low. We estimate the energy overhead introduced by the regulators due to voltage transition as:

$$E_{\text{regulator}} = (1 - \eta) \cdot C_{\text{filter}} \cdot |V_2^2 - V_1^2|$$  \hspace{1cm} (11)

where, $E_{\text{regulator}}$ is the energy dissipated by the voltage regulator due to a voltage transition, $\eta$ is the power efficiency of the regulator, $C_{\text{filter}}$ is the regulator filter capacitance, and $V_2$ and $V_1$ are the two voltage levels.

6. EXPERIMENTAL RESULTS

In this section, we evaluate the performance and temperature profiles of the mSWNoC and SWNoC, using the ALASH and AMROOTS routing and DVFS strategies elaborated in sections 4 and 5. For an exhaustive comparison, we also consider the performance and temperature profile of the conventional wireline mesh. The architecture of SWNoC is the same as that of the mSWNoC with long-range wired links instead of wireless shortcuts. We use GEM5 [Binkert et al. 2011], a full system simulator, to obtain detailed processor and network-level information. We consider a system of 64 alpha cores running Linux within the GEM5 platform for all experiments. The memory system is MOESI_CMP_directory, setup with private 64KB L1 instruction and data caches and a shared 64MB (1MB distributed per core) L2 cache. Three SPLASH-2 benchmarks, FFT, RADIX, LU [Woo et al. 1995], and two PARSEC benchmarks, CANNEAL and BODYTRACK [Bienia 2011], are considered. The processor-level statistics generated by GEM5 are incorporated into McPAT (Multicore Power, Area, and Timing) [Li et al. 2009] to determine the processor-level power statistics.

The width of all wired links is the same as the flit width, which is considered to be 32 bits in this paper. Each packet consists of 64 flits. Similar to the wired links, we have adopted wormhole routing in the wireless links too. The NoC simulator uses switches synthesized from an RTL level design using TSMC 65-nm CMOS process, using Synopsys™ Design Vision. The particular NoC switch architecture has three functional stages, namely, input arbitration, routing/switch traversal, and output arbitration. Each switch port has four virtual channels. Hence, four trees are created.
for AMROOTS and four layers are created for ALASH. All ports except those associated with the WIs have a buffer depth of two flits. The ports associated with the WIs have an increased buffer depth of eight flits to avoid excessive latency penalties while waiting for the token. Increasing the buffer depth beyond this limit does not produce any further performance improvement for this particular packet size, but will give rise to additional area overhead [Deb et al. 2012]. Energy dissipation of the network switches, inclusive of the rerouting block, was obtained from the synthesized netlist by running Synopsys™ Prime Power, while the energy dissipated by wireline links was obtained through HSPICE simulations taking into consideration the length and layout of the wireline links. In this work, we consider nominal range operation. Hence, the adopted DVFS strategy uses discrete V/F pairs that maintain a linear relationship. The considered levels are: 1V/2.5GHz, 0.9V/2.25GHz, 0.7V/1.75GHz, 0.6V/1.5GHz, and 0.5V/1.25GHz. As these voltages remain above threshold levels, they will not incur undesirably high performance degradation.

After obtaining processor and network power values, the processors and the network switches and links are arranged on a 20mm x 20mm die. These floor plans, along with the power values, are used in HotSpot [Skadron et al. 2003] to obtain steady state thermal profiles. The core powers and their corresponding network powers in presence of the specific benchmarks are fed to the HotSpot simulator to obtain the temperature profiles in each scenario.

### 6.1 Determination of the Small World Topology

We determine the exact topology of the SWNoC and mSWNoC based on the principles of the small-world graph as discussed in section 3. We then augment the network by adding WIs. Also from [Deb et al. 2012b], it is shown that WI placement is most energy-efficient when the distance between them is at least 7 mm in the 65 nm technology node. The optimum number of WIs is 12, for a 64-core system size [Wettin et al. 2013]. Increasing the number of WIs improves the connectivity of the network as they establish one-hop shortcuts. However, the wireless medium is shared among all the WIs and hence, as the number of WIs increases beyond a certain limit, performance starts to degrade due to the large token returning period [Deb et al. 2012b] and the energy dissipation starts to increase.

### 6.2 DTM/DVFS Setup

The DTM methodology requires tuning of the relevant parameters described in section 4 in order to optimize the achievable benefits. For congestion-aware routing in ALASH and AMROOTS, the monitoring window size, \( W_S \), requires tuning. By varying \( W_S \), we effectively define how fine-grained we want to monitor the traffic of the network switches. For AMROOTS the parameter for throttling threshold, \( \beta \), also needs to be tuned. By varying \( \beta \), we define how quickly we should begin to throttle potential hotspot switches. For each benchmark, relevant parameters were optimized in order to find the largest hotspot temperature reduction between the mesh and congestion-aware ALASH and AMROOTS routing strategies on the mSWNoC.

In order to optimize the DVFS methodology, three parameters, the prediction window (how often \( U_{predicted} \) is updated), the proactive switching window (how often the V/F can change for proactive DVFS), and the reactive switching window (how often the V/F can change for reactive DVFS) were varied between 100 and 1200 execution cycles. A small reactive switching window may catch data bursts, which do not represent a long-term trend of the benchmark’s traffic. Consequently, widely varying short-term traffic utilizations will cause the V/F to change often following the
reactive DVFS. In this case, the regulator energy overhead may outweigh the benefits of a lower misprediction penalty. As both reactive and proactive windows widen, the regulator energy overhead impact is decreased, while the latency penalty increases. For ALASH, the two DVFS prediction parameters, $\sigma$ and $\gamma$, were also optimized by individually varying the parameters to find the optimal configuration given a specific latency point. The case of increasing $\sigma$ causes $U_{\text{ALASH}}$ to follow $U_{\text{predicted}}$ while increasing $\gamma$ causes $U_{\text{ALASH}}$ to follow $U_{\text{Proactive}}$ more closely. By increasing the sum of $\sigma$ and $\gamma$ a more conservative prediction for the upcoming window can be achieved, which results in a lower latency penalty but less energy savings, while decreasing the sum of $\sigma$ and $\gamma$ results in a more aggressive prediction with a higher latency penalty but more energy savings.

6.3 Performance Evaluation

In this section, we present the latency, energy dissipation, and thermal characteristics of the mSWNoC with ALASH and AMROOTS routing by incorporating the joint DTM and DVFS techniques described earlier. For completeness, we also show the characteristics of the conventional wireline mesh and the SWNoC architectures incorporating the joint DTM and DVFS techniques.

6.3.1 DVFS Opportunities

The opportunity for performing DVFS on a NoC depends on the architecture’s link utilization characteristics. A histogram of the link-level traffic utilizations is shown in Fig. 3. Here, we highlight the differences between the mesh and mSWNoC to express the fact that the small-world architecture reduces link utilizations significantly. From this, it is seen that in the mesh architecture, a significant amount of links have more than 90% utilization for the considered benchmarks, and hence, the ability to perform link-level DVFS is low. In this case, there is not significant room for improvement with DVFS as the voltage and frequency cannot be tuned often on the links with high utilization without encountering excessive latency penalties. On the other hand, the mSWNoC reduces traffic on wireline links, which can also be seen clearly in Fig. 3. As the majority of links fall under 50% utilization in the mSWNoC architecture, there is significant opportunity for implementing DVFS. Because of this, there is room for more energy savings in mSWNoC in presence of DVFS compared to mesh. SWNoC also lowers link utilizations with respect to mesh. However, mSWNoC lowers it even further due to having wireless shortcuts. Also, differences between the routing strategies do not drastically affect these utilizations.

6.3.2 Latency and Energy Characteristics

Fig. 4 shows the latency for the various architectures and routing strategies. It can be observed from Fig. 4 that for all of the benchmarks considered, the latency of the small-world architectures with either routing strategies (ALASH and AMROOTS)
are lower than that of the mesh architecture. This is due to the small-world architecture of SWNoC and mSWNoC with direct long-range, one-hop wireline/wireless links that enables a smaller average hop-count than that of mesh. Between the mSWNoC routing strategies, ALASH performs better for all the benchmarks considered. This is due to the fact that ALASH is able to route through the shortest physical path while AMROOTS has to route through the tree, which may require a greater amount of hops, and also has the potential for congestion at the root nodes. This is shown in Fig. 5, which displays a normalized average traffic-weighted hop count of the various benchmarks for the two routing strategies on the mSWNoC. Here it can be seen that the routing in ALASH takes fewer hops to reach destinations over the AMROOTS routing strategy. Also, as the path selection is chosen to minimized congestion, there is further room for ALASH to improve in latency over AMROOTS when message conflicts start to occur in the network. With the addition of DVFS, a performance target that matches the latency of the wireline mesh was selected. Hence, implementing DVFS on mSWNoC will not introduce a performance impact with respect to the mesh. It can also be seen that by implementing DTM and DVFS on the mesh architecture its respective latency increases due to V/F misprediction and non-optimal path selection.

Fig. 6 shows the total network energy dissipation for the SWNoC, mSWNoC, and mesh architectures. It can be observed from Fig. 6 that in each benchmark, the

![Figure 4. Average network latency with various benchmarks.](image)

![Figure 5. Normalized average traffic-weighted hop count between routing strategies for mSWNoC.](image)
network energy is much lower for the small-world architectures compared to the mesh architecture. The two main contributors of the energy dissipation are the switches and the interconnect infrastructure. In the small-world architectures, the overall switch energy decreases significantly compared to a mesh as a result of the better connectivity of these architectures. In this case, the hop-count decreases significantly, and hence, on the average, packets have to traverse through less number of switches and links. In addition, a significant amount of traffic traverses through the energy-efficient wireless channels in mSWNoC; consequently allowing the interconnect energy dissipation of mSWNoC to further decrease compared to the SWNoC architecture. It can also be observed from Fig. 6 that the energy dissipation for ALASH and AMROOTS follows the same trend as that of the latency with ALASH having lower energy dissipation. When messages are in the network longer (higher latency) they dissipate more energy. The difference in energy dissipation arising out of the logic circuits of each individual routing is negligible and the overall energy dissipation is principally governed by the network characteristics.

With the addition of DVFS, the total network energy can be further reduced. As the traffic traversing through the wireline links is heavily reduced in the mSWNoC, which was discussed in section 4.2, the opportunity for implementing DVFS is significant. From Fig. 6 it is clear that for the CANNEAL benchmark mSWNoC saves 58.59% for ALASH and 55.84% for AMROOTS of energy with respect to the baseline mesh by incorporating only DTM. When the DTM is enhanced with DVFS the energy savings increases to 73.95% for ALASH and 58.82% for AMROOTS by allowing matching the original latency of the mesh architecture. It can be seen that by incorporating DVFS the energy dissipation advantage ALASH has over AMROOTS grows larger for the CANNEAL benchmark. For all benchmarks considered, ALASH has a lower latency compared to that of AMROOTS. This allows ALASH to have more margin than AMROOTS to match the latency of the original mesh. Hence, it is possible for ALASH to a more aggressive DVFS prediction without introducing latency penalty with respect to the baseline mesh, which in turn saves more energy. It is important to note that performing DTM and DVFS on the mesh architecture does not help in reducing the overall network energy. As the wireline mesh links are highly utilized, there is little opportunity to perform DVFS. For the two benchmarks with lowest switch interaction rates (RADIX and LU), implementing DVFS on the mesh architecture, allows for 7.8% and 4.6% energy savings, respectively, compared to the baseline mesh. In contrast, performing DVFS on a higher switch interaction rate benchmark like CANNEAL provides a 0.28% energy penalty as the regulator overhead consumes any savings in energy that was possible.
6.3.3 Thermal Characteristics

In this subsection we evaluate the thermal profiles of the mSWNoC, SWNoC, and mesh-based architectures employing DTM and DVFS techniques. The focus of this paper is to analyze the temperature profiles of the network components. However, we consider the effects of the processing cores in the HotSpot simulation to accurately portray the temperature-coupling effects that the processors have on their nearby network elements.

To evaluate the thermal profile of the network, we consider the improvement of maximum and average switch and link temperatures compared to the baseline mesh as $\Delta T_{\text{hotspot}}$ and $\Delta T_{\text{avg}}$ respectively as the two relevant parameters. Figs. 7 and 8 show these two parameters for the switches for the various architectures employing the DTM and DVFS techniques. Figs. 9 and 10 show these two parameters for the links for the various architectures employing the DTM and DVFS techniques. It can be seen that switches in the mSWNoC architecture are inherently much cooler (positive $\Delta T_{\text{hotspot}}$ and $\Delta T_{\text{avg}}$) than the mesh counterpart. From Fig. 6, we can see that the difference in energy dissipation between mSWNoC and mesh is significant and hence, it is natural that their switches are cooler. For the same reasons described earlier, the switches are cooler on the mSWNoC when compared to the SWNoC.

The benchmarks vary greatly in the switch interaction rates (injection loads). The FFT, RADIX, and LU benchmarks have very low switch interaction rates. While the CANNEAL and BODYTRACK have higher switch interaction rates than the others. Between the SWNoC and mSWNoC architectures, the SWNoC achieves a higher switch hotspot temperature reduction for the lower switch interaction rate benchmarks, as their traffic density is small. For these benchmarks, the benefits of the wireless shortcuts are outweighed by the amount of traffic that the WIs attract. However, for the higher switch interaction rate benchmarks, the use of high-bandwidth wireless shortcuts, in the mSWNoC, quickly relieves the higher amount of traffic that the WIs attract. In the case of SWNoC, as the shortcuts are implemented through multi-hop wireline links, moving traffic through these wireline links takes more time and energy which correlates with less temperature reduction.

It can be seen that the switches in the ALASH routing strategy are cooler than their AMROOTS counterpart. This follows the same trend seen in Fig. 6, ALASH has less energy dissipation which results in cooler switches. Performing adaptive routing on the mesh architecture is detrimental to the overall temperature profile, as the $\Delta T_{\text{hotspot}}$ and $\Delta T_{\text{avg}}$ for the majority of the benchmarks are negative (the mesh DTM scheme runs hotter than baseline mesh). As a large region of the mesh network already has hotspot issues [Wettin et al. 2013], adaptively rerouting traffic through...
extra hops only expands the hotspot region. However, the original hotspot switch temperature has in fact been reduced by 2.15°C in CANNEAL. However, while reducing the original switch hotspot temperature, a new switch has become hotter. The goal was to decrease the hotspot by routing away from the problem area, and mesh rerouting has in fact resulted in the opposite effect.

By performing DVFS on top of the congestion-aware routing schemes, the full thermal profile of ALASH and AMROOTS can be improved significantly. To compare the difference between the two routing strategies with their respective DVFS schemes, the difference in their hotspot temperatures ($\Delta T_{\text{ALASH-AMROOTS}}$) is considered as the relevant metric. For the switches, $\Delta T_{\text{ALASH-AMROOTS}}$ is 5.80°C, 7.10°C, 4.30°C, 13.97°C, and 9.87°C for FFT, RADIX, LU, CANNEAL, and BODYTRACK,

---

**Figure 8.** Average switch temperature reduction with DTM/DVFS techniques.

**Figure 9.** Hotspot link temperature reduction with DTM/DVFS techniques.

**Figure 10.** Average link temperature reduction with DTM/DVFS techniques.
respectively when matching the latency of the baseline mesh architecture. Hence, it can be seen that ALASH, while implementing DVFS, performs with less thermal hotspot switches than AMROOTS, as there is a larger margin to perform DVFS. Similar to the switches, the thermal profile of the links of ALASH and AMROOTS can also be improved through the use of DVFS. However, as seen by the switches, ALASH has the opportunity to perform a more aggressive DVFS approach. For the links, $\Delta T_{\text{ALASH-AMROOTS}}$ is $1.89^\circ\text{C}$, $2.77^\circ\text{C}$, $1.57^\circ\text{C}$, $5.61^\circ\text{C}$, and $3.38^\circ\text{C}$ for FFT, RADIX, LU, CANNEAL, and BODYTRACK, respectively at a comparable latency to the baseline mesh architecture.

In the next section, we explore the allowable latency penalty, with respect to mesh, in order to further lower the thermal profiles. Hence, depending on the performance requirements a suitable PTP trade-off can be established.

### 6.4 Design Space Exploration

As current systems are governed by many requirements of performance, power, heat, etc. there is an onus on the designer to decide what PTP is acceptable. In this respect, the DTM and DVFS algorithms discussed so far can be tuned to specific design requirements. The parameters that provide the optimum energy-delay product for given performance boundaries were found for the benchmarks considered in this work. Fig. 11 shows different DVFS operating points (various optimized parameters) for the CANNEAL benchmark as an example. In Fig. 11, the latency penalty, energy savings, and normalized energy-delay product are shown. In this work, so far we considered choosing the optimized DVFS parameters that provided no latency penalty with respect to the wireline mesh, which were the results presented in

**Figure 11.** Optimizing DVFS window sizes to fit performance boundaries vs. mesh for the CANNEAL benchmark for (a) AMROOTS and (b) ALASH.
earlier parts of this section. Further energy savings, and hence, temperature savings can be obtained by relaxing the acceptable performance boundaries. Given a specific application, a designer may choose an optimum DVFS configuration to suit their performance requirements while minimizing the energy dissipation of the network. From this figure, it can be seen that depending on design requirements, there is an adequate amount of room to obtain a large PTP trade-off. In the CANNEAL benchmark for example, if a maximum of 20% increase in latency is acceptable, then energy savings can be as much as 72.33%, with a latency penalty of 17.88% with respect to the baseline mesh architecture using the AMROOTS routing and DVFS methodologies. For the ALASH routing and DVFS methodologies that energy savings can increase to 84.12% with a latency penalty of 19.98% with respect to the baseline mesh architecture. Similarly, other trade-off points can be determined for the required PTP target. These trade-off points show how varying the associated window sizes affect the PTP. Furthermore, additional exploration of the ALASH parameters, $\gamma$ and $\sigma$, can be seen in Fig. 12. From this figure, it can be seen that varying these parameters slightly does not significantly affect the PTP, which is apparent by the first 12 bars that do not fluctuate significantly in latency, energy, or energy-delay product. However, as the sum of $\gamma$ and $\sigma$ is decreased below 90%, the prediction becomes more aggressive, which was described in section 6.2. Additionally, Table 1 shows the optimum parameter selections for the two routing strategies for DTM and DVFS on the mSWNoC architecture, where the blank table values indicate that those parameters are not used for that particular routing. This table shows that for ALASH, the lower utilization benchmarks (RADIX and LU) perform better when utilizing only the reactive DVFS. As the injection rate is decreased, which is the case for RADIX and LU, the amount of messages to predict is lower and hence $U_{\text{proactive}}$ is less reliable.

Table 1. Optimum parameter selections for mSWNoC.

<table>
<thead>
<tr>
<th>Application</th>
<th>$H_{\text{reactive}}$</th>
<th>$H_{\text{proactive}}$</th>
<th>$W_S$</th>
<th>$\beta$</th>
<th>$\sigma$</th>
<th>$\gamma$</th>
</tr>
</thead>
<tbody>
<tr>
<td>FFT</td>
<td>AMROOTS 100</td>
<td>100</td>
<td>250</td>
<td>0.022</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>ALASH 200</td>
<td>200</td>
<td>-</td>
<td>-</td>
<td>0.80</td>
<td>0.10</td>
</tr>
<tr>
<td>RADIX</td>
<td>AMROOTS 100</td>
<td>600</td>
<td>100</td>
<td>0.032</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>ALASH 1100</td>
<td>1100</td>
<td>-</td>
<td>-</td>
<td>1.50</td>
<td>0.00</td>
</tr>
<tr>
<td>LU</td>
<td>AMROOTS 100</td>
<td>1200</td>
<td>250</td>
<td>0.022</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>ALASH 800</td>
<td>800</td>
<td>-</td>
<td>-</td>
<td>2.00</td>
<td>0.00</td>
</tr>
<tr>
<td>CANNEAL</td>
<td>AMROOTS 200</td>
<td>100</td>
<td>100</td>
<td>0.027</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>ALASH 1200</td>
<td>1200</td>
<td>-</td>
<td>-</td>
<td>0.70</td>
<td>0.30</td>
</tr>
<tr>
<td>BODYTRACK</td>
<td>AMROOTS 100</td>
<td>600</td>
<td>100</td>
<td>0.050</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>ALASH 1100</td>
<td>1100</td>
<td>-</td>
<td>-</td>
<td>0.60</td>
<td>0.30</td>
</tr>
</tbody>
</table>

6.5 Area Overhead

The regulators were chosen for a fully distributed DVFS scheme where each wireline link and switch port has a private regulator (worst case). According to [Kim et al. 2008], the regulator area overhead for a single component capable of switching between the six V/F levels was computed to be 0.156mm². By considering a voltage...

Figure 12. Varying $\gamma$ / $\sigma$ vs. mesh for the CANNEAL benchmark for ALASH.
frequency island (VFI)-based design, this area overhead can be reduced, but is completely out of scope of this paper [Ogras et al. 2008]. Additionally, the hardware overhead for DVFS is shown to be 500 logic gates/switch/port, which is elaborated in [Shang et al. 2003a]. To measure switch and link utilizations, a counter at each output port gathers the total number of cycles that are used to relay flits in each history interval. These counters are reused to monitor the switch utilization for DTM, so that it does not introduce any additional area overhead. As this is small compared to the overall switch overhead, it is considered negligible for this work.

7. CONCLUSION

Millimeter-wave small-world wireless NoC (mSWNoC) is an enabling technology to design energy-efficient, high-bandwidth multicore architectures. The mSWNoC is capable of addressing the inherent performance limitations of conventional multi-hop wired mesh-based interconnection architectures. However, the gains in latency and power dissipation are not the only requirements for designing massive multicore chips; the thermal profile of these chips is also a very important parameter. As the power law connectivity-based mSWNoC is an irregular network topology, we need to design suitable congestion-aware routing strategies for this, which in turn influence the power and thermal profiles of the network. Therefore, the way the traffic is distributed around the mSWNoC through different congestion-aware routing strategies is important in reducing the network-level hotspots. In this paper we have demonstrated that by tightly coupling the congestion-aware routing mechanisms with suitable DVFS strategies, the power and thermal profiles of the mSWNoC can be improved significantly without introducing any performance degradation with respect to the baseline wireline mesh architecture. It has been seen that ALASH is able to distribute the traffic among the network better than AMROOTS. In this paper we demonstrate that due to the lower latency of ALASH, it is able to perform DVFS more aggressively when compared to AMROOTS. This results in an improved energy and thermal profile. In the CANNEAL benchmark, for example, the hotspot switch temperature was decreased by an additional 13.97°C and the energy was reduced by 36.7% for ALASH when compared to AMROOTS, with no additional latency penalty over the traditional mesh-based counterpart.

REFERENCES


