

# Thermal-aware task mapping on dynamically reconfigurable network-on-chip based multiprocessor system-on-chip

Liu, Weichen; Yang, Lei; Jiang, Weiwen; Feng, Liang; Guan, Nan; Zhang, Wei; Dutt, Nikil

2018

Liu, W., Yang, L., Jiang, W., Feng, L., Guan, N., Zhang, W., & Dutt, N. (2018). Thermal-Aware Task Mapping on Dynamically Reconfigurable Network-on-Chip Based Multiprocessor System-on-Chip. *IEEE Transactions on Computers*, 67(12), 1818-1834.  
doi:10.1109/TC.2018.2844365

<https://hdl.handle.net/10356/90107>

<https://doi.org/10.1109/TC.2018.2844365>

---

© 2018 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works. The published version is available at:  
<https://doi.org/10.1109/TC.2018.2844365>.

*Downloaded on 13 Mar 2024 16:41:22 SGT*

# Thermal-aware Task Mapping on Dynamically Reconfigurable Network-on-Chip based Multiprocessor System-on-Chip

Weichen Liu, *Member, IEEE*, Lei Yang, Weiwen Jiang, Liang Feng, Nan Guan, *Member, IEEE*, Wei Zhang, *Member, IEEE*, Nikil Dutt, *Fellow, IEEE*

**Abstract**—Dark silicon is the phenomenon that a fraction of many-core chip has to be turned off or run in a low-power state in order to maintain the safe chip temperature. System-level thermal management techniques normally map application on non-adjacent cores, while communication efficiency among these cores will be oppositely affected over conventional network-on-chip (NoC). Recently, SMART NoC architecture is proposed, enabling single-cycle multi-hop bypass channels to be built between distant cores at runtime, to reduce communication latency. However, communication efficiency of SMART NoC will be diminished by communication contention, which will in turn decrease system performance. In this paper, we first propose an Integer-Linear Programming (ILP) model to properly address communication problem, which generates the optimal solutions with the consideration of inter-processor communication. We further present a novel heuristic algorithm for task mapping in dark silicon many-core systems, called *TopoMap*, on top of SMART architecture, which can effectively solve communication contention problem in polynomial time. With fine-grained consideration of chip thermal reliability and inter-processor communication, presented approaches are able to control the reconfigurability of NoC communication topology in task mapping and scheduling. Thermal-safe system is guaranteed by physically decentralized active cores, and communication overhead is reduced by the minimized communication contention and maximized bypass routing. Performance evaluation on PARSEC shows the applicability and effectiveness of the proposed techniques, which achieve on average 42.5% and 32.4% improvement in communication and application performance, and 32.3% reduction in system energy consumption, compared with state-of-the-art techniques. *TopoMap* only introduces 1.8% performance difference compared to ILP model and is more scalable to large-size NoCs.

**Index Terms**—Reconfigurable NoC, Task Mapping, Communication Efficiency, Performance, Energy Consumption, Thermal reliability.

## 1 INTRODUCTION

CONTINUOUS scaling of technology leads to a utilization wall challenge in Network-on-Chip (NoC) based Multiprocessor System-on-Chip (MPSoC). Although more transistors can be packed in per area on the chip, the switching power per transistor is not scaling commensurately, hence the power density has been trending upwards. Coupled with the physical limits imposed by device packaging and cooling technology on peak power density, silicon chips cannot be fully utilized and a fraction of processor cores have to become powered-off or

underclocked in order to maintain the system allowable power budget and the chip temperature threshold, rising *Dark silicon* problem [1].

System-level thermal management techniques normally activate the physically decentralized cores such that the effect of thermal interaction can be reduced. Meanwhile, the heat dissipation can be enhanced since the active cores are surrounded by dark cores. Nevertheless, the main drawback of selectively activating cores in distributed locations is that the communication overhead is unexpectedly increased due to the longer average distance between active cores, when NoC is used as a scalable communication subsystem. The increased communication overhead would require processors running faster to mitigate the performance loss, which in turn increases the system energy consumption and chip power density. The contradiction between on-chip communication efficiency, system energy consumption and chip temperature is increasingly acute for systematical management in dark silicon many-core systems.

Distinct from traditional communication mechanisms, SMART [2][3], *Single-cycle Multi-hop Asynchronous Repeated Traversal*, was recently proposed to tailor a genetic fast NoC fabric at runtime. By traversing electrical signals across multiple hops via shared crossbar and links

\* Corresponding author: Lei Yang, College of Computer Science, Chongqing University, Chongqing, China. Department of Computer Science, University of California, Irvine, USA. Email: leiyang@cqu.edu.cn, lei.yang@uci.edu

W. Liu is with School of Computer Science and Engineering, Nanyang Technological University, Singapore. Email: liu@ntu.edu.sg

W. Jiang is with the College of Computer Science, Chongqing University, Chongqing, China, and with the Department of Electrical and Computer Engineering, University of Pittsburgh, Pittsburgh, PA 15261. Email: jiang.wwen@cqu.edu.cn; jiang.wwen@pitt.edu

L. Feng is with College of Computer Science, Chongqing University, Chongqing, China. Email: liangf@cqu.edu.cn

N. Guan is with Department of Computing, Hong Kong Polytechnic University. Email: csguan@comp.polyu.edu.hk

W. Zhang is with Department of Electronic and Computer Engineering, Hong Kong University of Science and Technology. Email: wei.zhang@ust.hk

N. D. Dutt is with Department of Computer Science, University of California, Irvine, USA. Email: dutt@ics.uci.edu

asynchronously within a single cycle, it enables long-distance message transmission through express channels with ultra-low latency. Matched flow-control and multi-hop setup techniques are proposed, pushing communication in NoCs to a nearly ideal framework with higher performance [4]. Dynamic reconfigurability of SMART NoC has given rise to the future scalability for kilo-core architectures to upgrade capabilities of software design.

Nevertheless, SMART can only support un-conflict messages to go through express bypass in intermediate routers, while conflicted messages still use traditional NoC for guaranteed delivery with extra delays. What's worse, flits should be stopped and buffered for arbitration in conflict routers, which will introduce extra overhead. In the extreme cases, no benefit can be acquired when flits are conflicted in each intermediate router. This will significantly degrade the effectiveness of SMART since flits cannot utilize one-cycle "hyperchannel" bypass effectively, and in turn reduces communication efficiency, especially for communication-intensive applications.

In this paper, we first model the entire problem by Integer-Linear Programming (ILP) formulations to find theoretically optimal solutions of task execution and communication. Then, we novelly propose a polynomial-time communication conflict- and thermal-aware task mapping algorithm, *TopoMap*, on top of SMART NoC architecture. NoC communication topology can be dynamically reconfigured in task mapping to minimize communication contention and maximize the utilization of single-cycle multi-hop bypass routing. Instead of conventional clustered approaches where cores have short distance, communicating tasks can be mapped to distant cores in SMART NoC as long as conflict-free communication paths can be established. Together with the consideration of chip thermal reliability, tasks mapped on physically non-concentrated cores can guarantee the safe chip temperature; meanwhile, these tasks will communicate in an ultra-low latency via express bypass channels dynamically built between active cores. Similar to software-defined network, *TopoMap* can control the reconfigurability of communication topology to reduce on-chip communication overhead and in turn improve system performance in thermal-safe guaranteed many-core systems. Systematical optimizations on-chip thermal reliability, communication performance, together with system energy efficiency, have been addressed in the proposed approaches on the reconfigurable SMART NoC. Evaluation results have verified the effectiveness of the ILP model and heuristic algorithm in communication efficiency, application performance, and system energy-performance efficiency compared with state-of-the-art techniques.

In the rest of the paper: Related work is discussed in Section 2; Section 3 presents the temperature-guaranteed task mapping; Section 4 demonstrates the effectiveness of contention-aware task mapping; Thermal-aware task mapping approaches in reconfigurable NoC are presented in Section 5; Experimental results and conclusion are given in Section 6 and Section 7, respectively.

## 2 RELATED WORK

**Communication efficient NoC design.** Since on-chip communication has become the bottleneck of NoCs, researchers devoted to improving communication efficiency. Physical express links and virtual channels (EVCs) are designed to tackle long-distance communication [5] [6], leading to the reduction in NoC diameter, thereby saving communication latency. Approaches in [7][8] can reconfigure NoC topologies and set up cross-bars at intermediate routers to reduce communication latency. Besides, on-chip communication efficiency has been further improved by addressing packets congestion through adaptive congestion-aware and contention-aware routing approaches [9][10][11][12]. For the new fabricated SMART NoC with the single-cycle multi-hop bypass routing, the on-chip routers are implemented in a characterized mechanism. The path control signals and the SMART-hop Setup Request (SSR) are conducted for setting up the express bypass routing, which is not similar to the conventional switch allocation in traditional NoC routers. Therefore, we have particularly studied the new features provided by SMART, and completely developed a matched policy for application mapping and scheduling, which has significantly enhanced the advantages of SMART NoC in communication efficiency.

**Application performance improvement.**

Contemporary researches of task mapping methods, which largely focus on data locality-centric application mapping to minimize the routing distance [13][14], are not applicable for the communication in SMART NoC. In conventional NoC platforms with the baseline router architecture, on-chip communication latency is determined by inter-processor distance and routing stages in intermediate routers based on hop-by-hop communication mechanism. Thus, the straightforward way to improve communication efficiency is to minimize the data transmission distance. Hence, data-dependent tasks are preferred to be mapped compactly [15][16]. With the consideration of the dynamic workloads in addition to the design-time task mapping, authors in [17][18][19] have investigated runtime application mapping heuristic algorithms to improve the system performance. While in SMART NoC, processor cores that are distributed on the chip with long routing distance can be selected for the allocation of communicating tasks since the long-distance communication can be realized in only one cycle via the single-cycle multi-hop bypass routing mechanism. While the communication contention could probably result in even larger percentage of overhead in the total performance since it takes time to establish the long-distance communication paths.

**Chip thermal reliability and system energy efficiency.** As chip thermal reliability and system power consumption have become the upcoming challenges in the optimization of MPSoCs [20], research efforts have made in [21][22][23][24][25] for thermal- and power-aware application mapping and scheduling. On top of it, the parallelism level of applications and running voltage/frequency values of processors will be determined at runtime to improve the overall

system performance and reduce energy consumption. Novel approaches based on folded torus and grouped processor core arrangement have been proposed with the matched power management policies in [26] and [27] to obtain power-performance tradeoff. Zhan *et al.* [28] had investigated a thermal-aware floorplan, on which processors can be sprinted with a power management scheme. Multiple layers of architecturally identical but physically different routers were integrated into the presented *darkNoC* architecture in [29]. Challenges had been addressed by Shafique *et al.* [1][30], including heterogeneous architecture synthesis and runtime power management, hoping to fully exploit the abundance of equipped transistors. However, with the new features provided by SMART NoC, matched techniques need to be presented to enhance the single-cycle multiple bypass routing technique, such that the communication efficiency, application performance, and chip thermal reliability can be further improved.

**Communication in SMART NoC.** Without extra physical links on SMART NoC [2], Low-swing clockless repeated link circuits embedded in router crossbars are driven asynchronously based on the electrical signal, supporting flits to traverse multiple hops to the destination in a single cycle. Kline *et al.* [31] presented a reservation based circuit-switching design, in which communication latency can be reduced by providing simplified global control. Specialized SSR (SMART-hop Setup Request) network is proposed to address the dedicated broadcast and the complex allocator for SSRs arbitration. SMART has enabled low latency by eliminating the delay in intermediate routers and delivers low power and area overhead. Nevertheless, the bypass will be interrupted where multiple packets are requiring the communication resource. Unfortunately, it will take several cycles for forwarding since flits should be stopped/buffered in contended routers. It will significantly degrade the effectiveness of SMART routers, especially for communication-intensive applications.

Take a panoramic view of the complicated situation for NoC design in dark silicon domain, there are threefold critical problems for task mapping and scheduling in many-core systems. First, how to decide thermal-aware task mapping on dark silicon patterns, such that the safe chip temperature can be guaranteed. Second, how to explore contention-free task mapping and configure the communication topology, in order to maximize the utilization of bypass. The third and the most crucial step is to determine thermo-reliable task mapping with contention-circumvented communication topology at runtime, which can satisfy both of the temperature and performance demands when targeting on the reconfigurable SMART NoC. In this paper, we propose ILP formulations to model communication in SMART NoC with fine-granularity, together with communication contention- and thermal-aware task mapping and scheduling algorithms. Objectives are to minimize communication conflict through task mapping and maximize bypass transmission, in order to improve application performance and energy efficiency with chip thermal reliability.

### 3 THERMAL-AWARE TASK MAPPING

We begin with the exploration of thermal-aware task mapping in dark silicon many-core systems. With the technology development, there is abundance of transistors packed on a single chip, such that they are available to be simultaneously activated to increase system performance. However, the power density has been trending upwards since the switching power for per transistor is not scaling commensurately, coupled with the physical limits imposed by device packaging and the cooling technology on peak power density and chip temperature. Consequently, on the contrast of the operation in traditional systems in which all cores can be powered-on at a nominal voltage/frequency level to boost computation efficiency under chip thermal constraint, silicon chip cannot be fully utilized that a large portion of transistors will be dark or dim at runtime, i.e., either be powered-off or under-clocked [1][30]. Therefore, which and how processor cores can be selected and activated in the dark silicon patterns will exhibit multitude power modes and in turn result in starkly distinct thermal profiles [28][32].

#### 3.1 Thermal-aware Dark Silicon Pattern

The goal is to decide dark silicon pattern in which portion of cores will be powered on for computation, meanwhile, to save system energy consumption and maintain safe chip temperature. As illustrated in Fig. 1, we have conducted an experiment to illustrate the resulted thermal distributions after a period of computation on different dark silicon patterns. In order to show the different chip thermal profiles resulted by different dark silicon patterns, in this motivational example, we simply assume there are 50% cores on the chip will be dark and others will be activated in different patterns. In each setting, a same set of multi-thread applications (H.264) are executed on active cores at a full voltage/frequency level. We model per-core computation power by McPAT [33] at 22-nm node technology. Chip temperature and the steady state thermal maps are obtained by *HotSpot v 5.02* [34] with the set temperature threshold  $81.8^\circ\text{C}$  [35].

We can observe that: i) The naive approach is to activate a set of contiguous processors [36], as pattern 1, 2 and 3. There is no doubt that it will lead to serious regional temperature hot spots induced by dense power consumption. What is worse, chip peak temperature generated in the center of the powered-on area (reaching up to  $82.17^\circ\text{C}$ ,  $82.15^\circ\text{C}$ ) tends to rise perpendicularly, which has violated safe temperature threshold  $T_{safe} = 81.8^\circ\text{C}$  [35]). ii) Selectively activate processors in random patterns, pattern 4 and 5, have alleviated power density

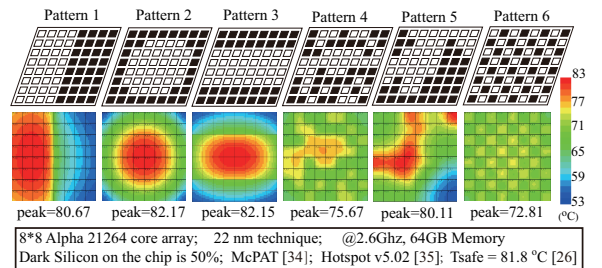


Fig. 1: Thermal profiles of different dark silicon patterns.

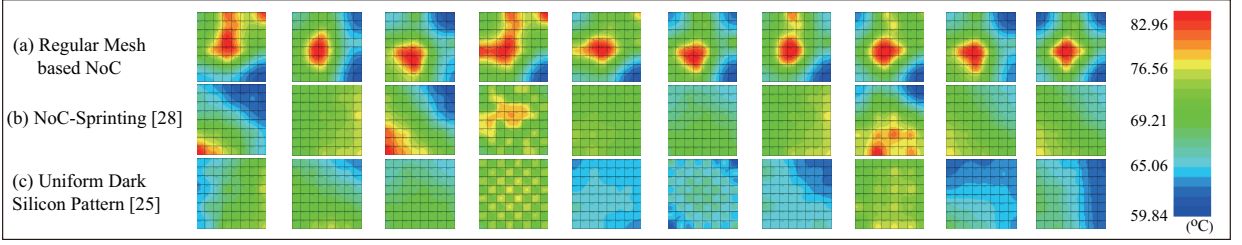


Fig. 2: Comparison of the heat distribution by mapping and scheduling a set of PARSEC benchmark applications on (a) Mesh-based NoC; (b) NoC-Sprinting [28]; (c) Uniform dark silicon pattern [25].

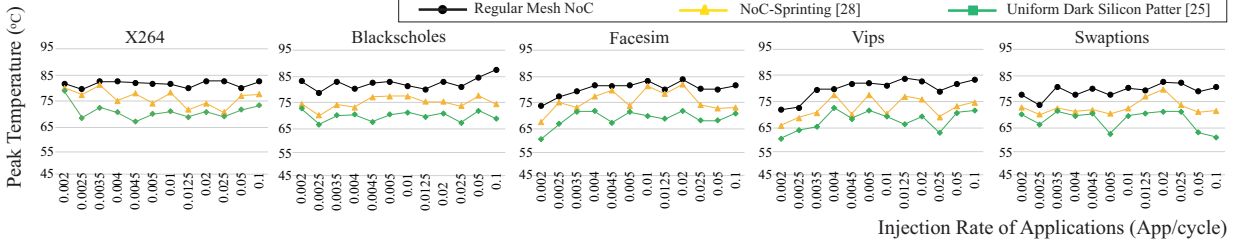


Fig. 3: Comparison of chip peak temperature ( $T/^{\circ}\text{C}$ ) by mapping and scheduling a set of PARSEC benchmark applications on (a) Mesh-based NoC; (b) NoC-Sprinting [28]; (c) Uniform dark silicon pattern [25].

problems to a certain extent due to spatial heat dissipation. As a result, chip peak temperature can be reduced by  $6.5^{\circ}\text{C}$  compared with that in contiguous patterns. Nevertheless, it is inevitable to activate adjacent cores in regions that it is hard to find consistent optimal solutions with random method. As the thermal maps demonstrated in Fig. 1, the random pattern 5 has caused hot spots in the left side and the top right corner of the chip. iii) In pattern 6, active cores are evenly distributed on the chip, surrounding with dark cores in each direction. As confirmed in [25], the produced power is configured in a uniform layout, which can effectively avoid regional hot spot. In addition, four dark cores around are beneficial for heat dissipation. As a consequence, chip peak temperature can be  $9.3^{\circ}\text{C}$  lower, which is a persistent temperature returns.

In order to obtain the general results, we have conducted a set of experiments by deadline constraint application mapping and scheduling in different core organizations, including mesh-based NoC, NoC-Sprinting [28] and the symmetrically distributed dark silicon pattern [25], for comparing the thermal conditions and the chip temperature. As demonstrated in Fig. 2 and Fig. 3, we have obtained the thermal maps of the steady state heat distribution and the chip peak temperature generated by three approaches after running a set of PARSEC benchmark applications [37].

Generally, tasks are mapped next to each other to achieve better performance, such that heat generated by running cores will be concentrated within an area, which induces serious chip hot spots. Actually, increased high chip temperature will threaten system reliability. Thus, these cores on mesh cannot be simultaneously powered on in real systems. In NoC-Sprinting, fine-grained sprinting has mitigated hot spots via thermal-aware floorplanning. However, a fraction of contiguous cores should be activated to reduce communication latency. Thus, NoC-Sprinting can only result in a little more moderate heat accumulation than that on mesh. Thermal gradients are effectively reduced in the uniform dark silicon pattern,

on which tasks are physically distributed on chip in a well-proportioned configuration, resulting in much better chip profiles as the thermal maps exhibited in Fig. 2. As a consequence, chip peak temperature can be reduced on average  $9.13^{\circ}\text{C}$  and  $4.02^{\circ}\text{C}$ , respectively, compared with that on mesh and NoC-Sprinting as illustrated in Fig. 3.

### 3.2 Temperature Guaranteed Task Mapping

Uniform dark silicon pattern can effectively manage thermal and reduce chip temperature in terms of the special physical arrangement. In general, patterns with different darkness ratio can be theoretically determined by Fourier's law:  $q = -kA \frac{dT}{dx}$ , by considering the conductance between two heatsink elements [38].  $q$  is heat flow;  $k$  is thermal conductivity;  $A$  is cross sectional area;  $dT$  is temperature difference between objects and  $dx$  is the distance between objects. The law of heat conduction states that time rate of heat transfer through a material is proportional to the negative gradient in temperature and to the area, at right angles to that gradient, through which heat flows. It provides the definition of thermal conductivity and shows that heat flow has an inverse relationship with physical distance of objects. Simply understand, heat conduction is more serious with surrounding objects which have a shortest Euclidean Distance.

For any two communicated nodes, mapped on specific cores,  $\mathcal{M}_{v_i \rightarrow p_i} = 1$ ,  $\mathcal{M}_{v_j \rightarrow p_j} = 1$ , with coordinate  $p_i(x_i, y_i)$ ,  $p_j(x_j, y_j)$ , their Euclidean Distance ( $\mathcal{ED}$ ) is:

$$\mathcal{ED}_{p_i, p_j} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \quad (1)$$

In order to relieve the regional heat accumulation and prevent the resulted hot spot induced by contiguous task mapping, it seems that activate cores with relatively larger Euclidean Distance on the physical chip would be a better choice. Nevertheless, there will not be such adequate processors available for the increasing computing tasks [25][32]. When temperature is considered as a constraint in estimating dark silicon, it allows only a fraction



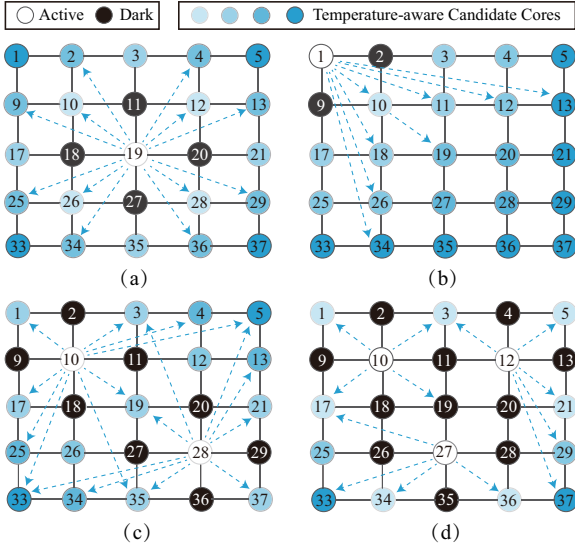


Fig. 4: Tasks mapping on multiple candidate cores.

of cores activated in most of the cases. According to the primary criterion of Fourier's law, tasks will be mapped on distributively activated processors on physical floor-plan to maintain the thermo-reliable chip profile and keep system temperature under the constraint. Without loss of generality, we first assume that processors next to any active cores will prefer to be dark, as core 11, 18, 20, 27 and core 2, 9 illustrated in Fig. 4 (a)(b).

As the motivational example shown in Section 3.1, the dark silicon pattern with processor cores organized as Pattern 6 can better relieve the heat accumulation than other patterns. Thus, in our presented temperature-aware mapping method, processor cores in physically decentralized locations surround by dark cores will be activated, such that the power density can be reduced with a resultant lower chip temperature, and in turn to improve the thermal reliability.

In general, real-time running circumstance will be more complicated since there will be continual released applications to be assigned on the profile with a set of currently active cores. In these cases, we should make a decision to select proper cores among candidates for task assignment without violating temperature constraint as shown in Fig. 4 (c)(d). With the objective to diminish power density and provide a best possible thermo-friendly chip profile for new applications, as described in Algorithm 1, **Temp\_Map**, tasks will be mapped to cores which have a *maximized minimum distance* between all of currently active cores, that is  $\max\{\min\{\mathcal{ED}(p_j, p_k)\}\}$ .

### 3.3 Task Mapping Distance Lower Bound: $B_{Lower}$

Chip thermal reliability can be maintained in a better heat condition benefiting from the interval arrangement of active/dark cores. In this dark silicon pattern, an active core is surrounded by four dark cores within one hop. Notice that, according to the technological data from ITRS and Intel, at 8-nm node, more than 70% of the chip will be dark [30]. Thus, in this paper, we will powered-off cores within one hop from an active core as demonstrated in Fig. 4, in order to eliminate the direct effect of heat conduction from neighboring active cores. In other words, tasks will be restrictively mapped to cores which are at

## Algorithm 1 Temperature-aware Task Mapping

---

**Temp\_Map** ( $v, p, \mathcal{P}_A$ )

---

```

1: for  $v_i \in \mathcal{V}$  do
2:   for  $p_j \in (\mathcal{P} - \mathcal{P}_A)$  do
3:      $\mathcal{T}_{v_i} = \text{MatEx}(\mathcal{M}_{v_i \rightarrow p_j})$ ; /* Obtained by MatEx [39] */
4:     if  $\mathcal{T}_{v_i} < \mathcal{T}_{safe}$  then
5:       for  $p_k \in \mathcal{P}_A$  do
6:          $\mathcal{ED}(p_j, p_k) = \sqrt{(x_j - x_k)^2 + (y_j - y_k)^2}$ ;
7:          $\text{Thermo\_ED} = \max\{\min\{\mathcal{ED}(p_j, p_k)\}\}$ ;
8:         Break;
9:       end for
10:      Return  $\mathcal{M}_{v_i \rightarrow p_j} = 1$ ;
11:     end if
12:   end for
13: end for

```

---

least 2-hop away from active cores as long as the safe temperature can be kept. This will be broken that adjacent cores can be selected if and only if there are no available cores which have longer distance than 2 hops. Instead of exploring all cores on the chip for task mapping, the search space can be significantly reduced in our approach.

## 4 COMMUNICATION-AWARE TASK MAPPING

As stated in previous sections, we unilaterally optimize chip peak temperature by mapping tasks on physically decentralized cores. However, the main drawback is the unexpected increase of the inter-processor communication overhead due to the averagely longer distance between active cores, when traditional NoC topology (e.g., mesh) is implemented as the communication subsystem. Actually, experimental results show that the communication performance of the uniform dark silicon pattern 6 is on average 37.4% worse than that of pattern 1, 2 and 3 in Fig. 1. The loss of communication performance would be much worse for communication intensive applications. Anyway, in this section, we focus on the optimization of on-chip communication efficiency and present a novel task mapping algorithm on the reconfigurable NoC with the consideration of communication contention to reduce long-distance communication latency.

### 4.1 Background

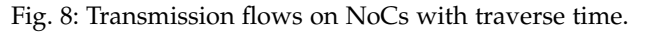
**Communication in Regular NoC.** In conventional NoCs, communication is conducted by hop-to-hop router + link traversal, where on-chip routers perform multiplexing of flits on the shared links. A router will be responsible for the actions including: *Buffer Write (BW)*, *Route Compute (RC)*, *Switch Allocation (SA)*, *Virtual Channel Selection (VS)*, *Switch Traversal (ST)* and *Link Traversal (LT)*. Generally, the latency for a packet transmission is linear with the distance  $h_{path}$  and the routing stages in the routers, which can be calculated as:

$$L_{comm} = (L_r + L_w) \times h_{path} + (W_{flit} - 1)/b \quad (2)$$

where  $L_r$  is the number of router stages (e.g., 2 or 3 in traditional router).  $L_w$  represents the delay on link between two routers.  $W_{flit}$  is denoted as the number of flits and  $b$  is the bandwidth of link channels.

Except the work to reduce the long-distance between the processors, researchers [40] have made efforts to improve the communication efficiency on regular NoCs by

In this work, in order to maximize the utilization of



We continue to give an example of application mapping and scheduling in regular and SMART NoC to show the effect of flits conflict on communication efficiency and to illustrate the effectiveness of SMART NoC. Given a task graph, which is consisted of six nodes  $v_1$  to  $v_6$ , task

execution times are 4, 4, 2, 12, 8 and 16, and communications are  $v_0 \rightarrow v_2 = 16$ ,  $v_1 \rightarrow v_2 = 16$ ,  $v_2 \rightarrow v_4 = 10$ ,  $v_3 \rightarrow v_4 = 10$  and  $v_3 \rightarrow v_5 = 20$ , respectively. We map tasks on 2-D mesh-based NoC (with regular 3-stage router) and SMART NoC (express bypass channel in the router) by methods in Davare [14], Yu [16] and our strategy in Section 4.3. On top of it, we obtain schedules of task execution (orange bar) and communication (gray bar) based on XY-routing as exhibited in Fig. 9.

We can observe that schedules on SMART NoC by three methods are better than their respective results on regular NoC. Besides, mappings by Davare [14] and Yu [16] on either regular NoC or SMART NoC have longer schedule length than our approach due to communication conflicts. On regular NoC, the transmission of contended packets are delayed, such that the schedule length will be stretched. Likewise, bypass cannot be successfully carried out in above two mappings when communication conflict occurs, thus extra overhead is introduced. Our mapping with fine-grained consideration of contention (detailed in Section 4.3) can obtain the shortest schedule length both on regular NoC and SMART NoC, as shown in Fig. 9 (c), since there is no contention and all packet transmissions can go through bypass in intermediate routers.

There are threefold inspirations indicated by this example: i) SMART NoC have enabled the ultra-low latency via 1-cycle bypass through intermediate routers entirely, which makes great progress in on-chip communication efficiency. Task schedule on SMART NoC can significantly reduce the schedule length compared with the respective results in regular NoC. ii) In order to maximize the benefits of SMART design and enhance its effectiveness in

communication, the conflict induced overhead should be minimized with the maximized utilization of bypass routing. Mappings [14][16] on either regular NoC or SMART NoC have longer schedule length than our approach due to communication conflicts. iii) When considering chip thermal reliability, the chip peak temperature would be much worse in the mappings by [14][16] since tasks are mapped and executed centralized on the chip.

### 4.3 Communication Contention-aware Task Mapping

Based on the observation, in SMART NoC, single-cycle multi-hop bypass channel will be built on account of runtime communication patterns, which are related to task mapping and communication circumstances. In this work, we novelly propose a system-level communication contention-aware task mapping approach, which can control the configuration of logical NoC interconnection dynamically and construct communication topology, to maximize the utilization of bypass and in turn to improve communication efficiency. The same manufacture as software defined network (SDN) [43], our proposed task mapping is the software which can control and optimize the performance of the reconfigurable NoC. Without too much concern about the physical distance between distant processors, communicated tasks can be mapped non-contiguous to evade spatial conflict and enable more bypass routing in SMART NoC, instead of the conventional communication gathered task mapping.

We first present an ILP model for task mapping and scheduling in SMART NoC. Solutions are composed of task mapping, execution and communication scenarios via SMART router. In Table 1, we have summarized the notations used for constructing ILP formulation with detailed definition. In addition, six primary Boolean variables are introduced to simplify the problem formulation. In this ILP model, we simply assume that wherever there is more than one shared link in two packets transmission routes, communication conflict occurs. Realistic communication scenarios will be much better than that in this model, since pairs of contended routes may have no transmission time overlap on the shared links.

**(1) Basic task execution constraints:** Each task is mapped to a processor and invoked the precedence relations.

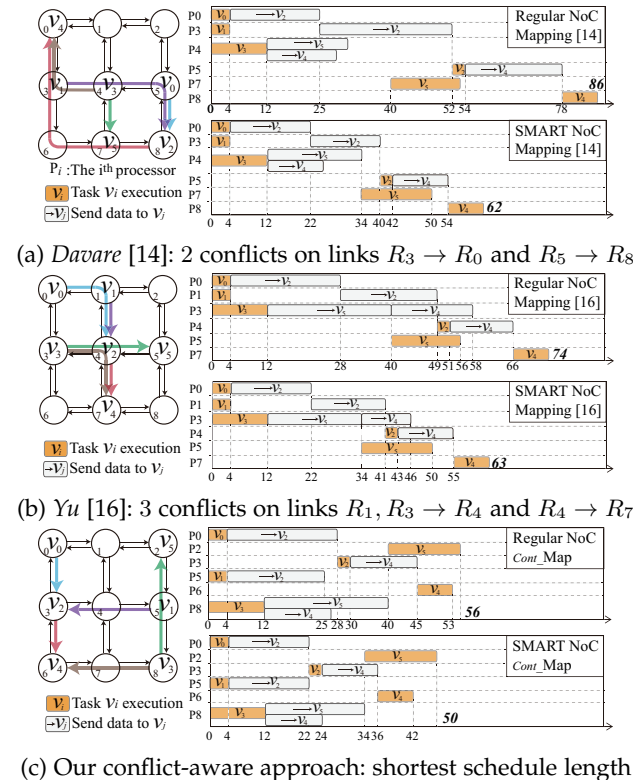


Fig. 9: Mapping and scheduling by approaches on regular NoC (3-stage router) and SMART NoC (1-cycle bypass).

TABLE 1: Notations Used in ILP Formulations.

Notation	Definition
$\mathcal{V}, \mathcal{E}, \mathcal{P}$	The set of task nodes, edges and processors.
$\tau_{u,p}$	The execution time of task $u$ on processor $p$ .
$s_u / f_u$	Release/Completion time of task $u$ .
$\mathcal{W}_{u,v}$	The packet size on edge $e_{u,v}$ .
$s_{u,v} / f_{u,v}$	Start/Completion time of message on $e_{u,v}$ .
$h_{p,q}$	The Manhattan Distance from processor $p$ to $q$ .
$f_{app}$	Completion time of the entire application.
$\mathcal{N}$	A large positive integer constant, and $\mathcal{N} > \max\{f_u\}$ .
$\mathcal{T}_{safe}$	The safe chip temperature.
$HPC_{max}$	The maximum hop flits can be traversed per cycle.
Notation	Boolean Variables
$\mathcal{M}_{u,p}$	= 1, iff task $u$ is mapped to processor $p$ .
$o_{u,v}$	= 1, iff task $u, v$ have execution overlap.
$\ell_{u,v}$	= 1, iff $u$ starts serially after task $v$ completed.
$\alpha_{e_i,e_j}$	= 1, iff $e_i, e_j$ have shared virtual link(s).
$\beta_{e_i,e_j}$	= 1, iff $e_i, e_j$ have transmission time overlap.
$\gamma_{e_i,e_j}$	= 1, iff $\alpha_{e_i,e_j} \wedge \beta_{e_i,e_j} = 1$ & $Prio(e_i) = high$ .



$$\sum_{p \in \mathcal{P}} \mathcal{M}_{u,p} = 1, \quad \forall u \in \mathcal{V}. \quad (4)$$

$$f_u = s_u + \sum_{p \in \mathcal{P}} (\tau_{u,p} \times \mathcal{M}_{u,p}), \quad \forall u \in \mathcal{V}, \quad \forall p \in \mathcal{P}. \quad (4)$$

$$s_v \geq f_u, \quad \forall e(u, v) \in \mathcal{E}.$$

**(2) Basic message transmission constraints:** The minimum message transmission latency on regular NoC.

$$s_{u,v} \geq f_u, \quad s_v \geq f_{u,v}, \quad \forall u, v \in \mathcal{V}. \quad (5)$$

$$f_{u,v} \geq s_{u,v} + 3 \times (\mathcal{M}_{u,p} + \mathcal{M}_{v,q} - 1) \times (h_{p,q} + 1) + (\mathcal{W}_{u,v} - 1) - (1 - \mathcal{M}_{u,p}) \times \mathcal{N} - (1 - \mathcal{M}_{v,q}) \times \mathcal{N}, \quad (6)$$

$$\forall u, v \in \mathcal{V}, \quad \forall p, q \in \mathcal{P}.$$

**(3) Task overlap constraints:** For tasks  $u, v \in \mathcal{V}$  which are neither in each other's *transitive fan-out (TFO)*, the first constraint ensures if  $u$  finishes after  $v$  begins, the corresponding variable  $o_{u,v}$  is set to 1. The second constraint ensures no two tasks mapped on a processor can overlap. Thus,  $o_{v,u} + o_{u,v} = 2$  iff  $u, v$  have execution overlap. For all other cases, the sum of  $o_{v,u}$  and  $o_{u,v}$  must be 1.

$$f_u - s_v \leq \mathcal{N} \times o_{u,v}, \quad \forall u, v \in \mathcal{V}. \quad (7)$$

$$o_{v,u} + o_{u,v} + \mathcal{M}_{u,p} + \mathcal{M}_{v,p} \leq 3, \quad \forall p \in \mathcal{P}.$$

**(4) Message overlap constraints:** For two messages on  $e_i = (u_i, v_i), e_j = (u_j, v_j) \in \mathcal{E}$ , communication overlap can be formulated as task overlap constraints.

$$f_{u_j, v_j} - s_{u_i, v_i} \leq \mathcal{N} \times \gamma_{e_i, e_j}, \quad \forall u, v \in \mathcal{V}. \quad (8)$$

$$\gamma_{e_i, e_j} + \gamma_{e_j, e_i} + \alpha_{e_i, e_j} + \beta_{e_i, e_j} \leq 3, \quad \forall e_i, e_j \in \mathcal{E}.$$

**(5) Communication contention in time:** For any two messages on edges  $e_i, e_j \in \mathcal{E}$ , the following set of constraints defines time overlap of the two messages.

$$(s_{u_i, v_i} \leq f_{u_j, v_j}) \wedge (f_{u_i, v_i} \geq s_{u_j, v_j}), \quad (9)$$

$$(s_u - f_v) / \mathcal{N} \leq \ell_{u,v} \leq (s_u - f_v) / \mathcal{N} + 1,$$

$$\ell_{v_j, u_i} / 2 + \ell_{v_i, u_j} / 2 - \lambda \leq \beta_{e_i, e_j} \leq \ell_{v_j, u_i} / 2 + \ell_{v_i, u_j} / 2,$$

$$\forall u, v \in \mathcal{V}, \quad \forall e_i, e_j \in \mathcal{E}.$$

where  $\lambda$  is a constant for linearization, and set as 0.75.

**(6) Communication contention in space:** For two edges  $e_i = (u_i, v_i), e_j = (u_j, v_j)$ , we denote processors by coordinates where tasks  $u_i, v_i, u_j$  and  $v_j$  are mapped, that  $u_i = (x_{u_i}, y_{u_i}), v_i = (x_{v_i}, y_{v_i}), u_j = (x_{u_j}, y_{u_j})$  and  $v_j = (x_{v_j}, y_{v_j})$ .  $\alpha_{e_i, e_j} = 1$  represents that  $e_i, e_j$  have at least one shared link. In other words, transmission paths have overlap in space.  $\mathcal{M}_{v \rightarrow p} = 1$  refers that task  $v$  is mapped to processor  $p$ . To accurately model it, contended links will be in  $x$ - or  $y$ -coordinate, when NoC is regarded as a two-dimensional coordinate axis. In the former case, shared links in horizontal direction must have a same  $y$ -coordinate of their sources as formulated in Equation (10).

$$\alpha_{e_i, e_j} = [\mathcal{M}_{u_i \rightarrow p} + \mathcal{M}_{v_i \rightarrow q} + \mathcal{M}_{u_j \rightarrow m} + \mathcal{M}_{v_j \rightarrow n} - 4] + ((y_{u_i} = y_{u_j}) \wedge \{ [(x_{u_i} > x_{u_j}) \wedge (x_{u_i} < x_{v_j}) \wedge (x_{u_i} < x_{v_i})] \vee [(x_{u_i} < x_{u_j}) \wedge (x_{u_j} < x_{v_i}) \wedge (x_{u_j} < x_{v_j})] \vee [(x_{u_i} > x_{u_j}) \wedge (x_{u_j} > x_{v_i}) \wedge (x_{v_j} < x_{u_j})] \vee [(x_{u_i} < x_{u_j}) \wedge (x_{u_i} > x_{v_j}) \wedge (x_{v_i} < x_{u_i})] \})). \quad (10)$$

The other case, in which two paths have shared link(s) in vertical direction, can be similarly formulated.

**(7) SMART communication delay constraints:** Communication delay is upgraded as described in Section 4.

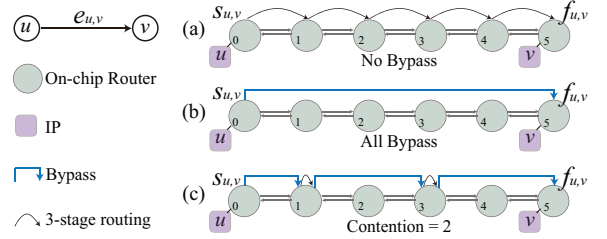


Fig. 10: Communication through SMART routers.

$$f_{u,v} \geq s_{u,v} + 3 \times (\mathcal{M}_{u,p} + \mathcal{M}_{v,q} - 1) \times \sum_{e_j \in \mathcal{E}} \gamma_{e_i, e_j} + 3 \times 2 + (\sum_{e_j \in \mathcal{E}} \gamma_{e_i, e_j} + 1) + (\mathcal{W}_{u,v} - 1), \quad \forall e_i, e_j \in \mathcal{E}, \quad \forall p, q \in \mathcal{P}. \quad (11)$$

where  $3 \times 2$  represents the delay in source and destination routers as shown in Fig. 10.  $(\sum \gamma_{e_i, e_j})$  is the sum of conflicts, and  $(\sum \gamma_{e_i, e_j} + 1)$  is the number of bypass.

**(8) Objective:** Minimize application schedule length.

$$f_{app} \geq f_u, \quad \text{Min}\{f_{app}\}, \quad \forall u \in \mathcal{V}. \quad (12)$$

The model is established to illustrate mapping and scheduling with detailed communication scenarios for regular/SMART NoCs. It is compatible with any deterministic routing algorithms as long as the transmission path can be decided at design-time. Nevertheless, this model can incorporate other routing algorithms by modifying Equation (10) accordingly to address spacial contention. Besides, Equation (5)(11) can be revised for distinct realization of router mechanisms.

Since the mapping problem is NP-hard and the design space is exponentially increased, the running time of ILP rises rapidly with the increasing size of applications [44]. It is not surprising that our technique has introduced a new dimension to detect contention in search space, which might be huge since edges have a quadratic relation to tasks. To effectively solve the complex issue and analyze communication contention in SMART NoC, we propose polynomial-time heuristic algorithms for contention-aware task mapping. We can simply divide two types of data paths as illustrated in Fig. 11. In order to avoid transmission overlap in spacial dimension, dependent tasks having possibilities of conflict will be mapped to distinct areas. Generally, there are two cases:

**Case-1** Packets on edges  $e_{v_i, v_k}$  and  $e_{v_j, v_k}$  join to a same node. Data paths, generated by XY-routing, have the same destination router. As shown in Fig. 11 (a), the ideal mappings of tasks  $v_i, v_j$  and  $v_k$  are that: if  $\mathcal{M}_{v_k \rightarrow p_{19}} = 1$ ,  $v_i$  can be firstly mapped to one of the processors in the bottom gray area where the  $y$ -coordinate is not larger than  $y_{v_k}$ ,  $y_{v_i} \leq y_{v_k}$ . Then  $v_j$  will be mapped to the processor cores in the upper area marked as pink, where  $y_{v_j} > y_{v_k}$ . **Case-2** Packets on edges  $e_{v_i, v_k}$  and  $e_{v_j, v_k}$  are produced from a node. Their data paths are rooted from a same source. As shown in Fig. 11 (b), the ideal contention-free mapping of tasks  $v_i, v_j$  and  $v_k$  are that: if  $\mathcal{M}_{v_k \rightarrow p_{19}} = 1$ ,  $v_i$  can be firstly mapped to one of the processors in the left gray area, where the  $x$ -coordinate is not larger than  $x_{v_k}$ ,  $x_{v_i} \leq x_{v_k}$ . Thus,  $v_j$  will be mapped to the processor cores in the left area marked as pink, where  $x_{v_j} > x_{v_k}$ .

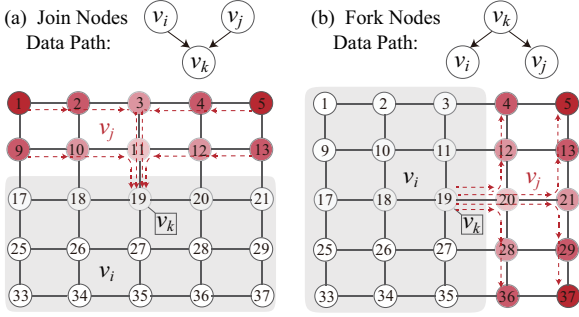


Fig. 11: Task mapping on SMART NoC with consideration of spatial communication contention.

Communication conflict in spacial dimension will be reduced in task mapping by Algorithm 2, **Cont\_Map**, thus communication latency can be significantly decreased. As schedule lengths obtained by ASAP (As Soon As Possible) on DSP-stone [45] benchmarks in Fig. 12 (a)(b)(c), Algorithm 2 can significantly reduce communication latency in SMART NoCs due to the minimized communication contention, on average 40.1%, 42.7% and 43.3%, respectively, compared with that obtained by Davare [14], N.Koziris [15] and Yu [16]. In addition, with increased mesh size, we can obtain better results as demonstrated in Fig. 12 (d), since more cores are available to be selected in task mapping, thus communication conflict can be further reduced.

#### Algorithm 2 Contention-Circumvented Task Mapping

```

Cont_Map ( $v, p$ )
1:  $Q \leftarrow$  Initial queue for  $v \in \mathcal{V}$  as decreased out-degree;
2: for  $v_i, v_j, v_k \in Q$  do
3:   if  $(e_{v_k, v_i} = 1) \wedge (e_{v_k, v_j} = 1) \wedge (\mathcal{M}_{v_k \rightarrow P_k} = 1) \wedge (\mathcal{M}_{v_j \rightarrow P_j} = 1)$ 
     then
4:      $Content(e_{k,i}, e_{k,j}) = \alpha_{e_{k,i}, e_{k,j}}$ ; /* Equation (10)(??) */
5:     if  $Content(e_{k,i}, e_{k,j}) = 0$  then
6:        $Return \mathcal{M}_{v_i \rightarrow P_i} = 1$ ; Break;
7:     end if
8:   end if
9: end for

```

#### 4.4 Task Mapping Distance Upper Bound: $B_{Upper}$

It is contention matters, not the distance, in task mapping on SMART NoC. In the contention-aware task mapping approach, distant task mapping method can explore more available cores with the objective to avoid communication conflict, such that the utilization of bypass can be effectively enhanced coupled with the reduced communication latency. Nevertheless, it is not an absolute criterion to map tasks on cores as far as possible until the contention-circumvented routing paths are built due to two reasons.

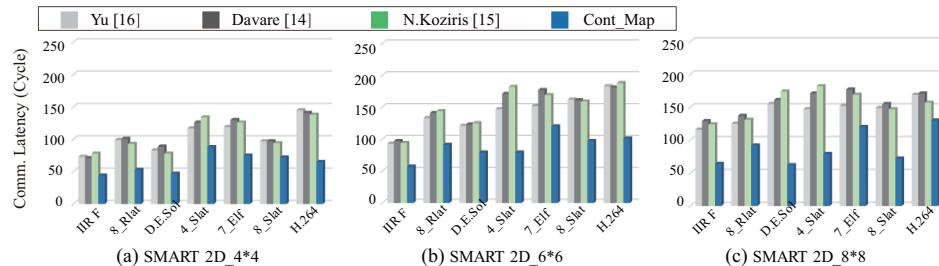


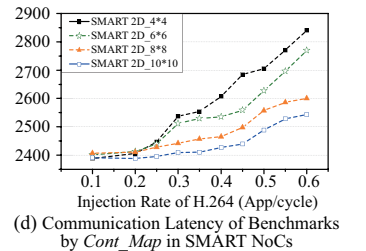
Fig. 12: Comparison of communication latency by different mapping methods on (a)  $4 \times 4$ , (b)  $6 \times 6$  and (c)  $8 \times 8$  SMART NoCs. (d) Scalability of the communication contention-aware task mapping approach.

First, SSR is allowed to broadcast to the cores in a distance from 1 hop to  $HPC_{max}$ , which is the maximum hop that flits can be traversed per cycle. Even though a bypass is enabled, flits have to stop in the  $HPC_{max}$  router before reaching the destination [42]. That is to say, if inter-processor distance (Manhattan Distance) is longer than the maximum hop that bypass can traverse to, there will be at least one stop in the data path. Second, since communication resources are limited on NoC, data path configured across long distance will relatively occupy more router ports and link channels, which will lead to more potential transmission conflicts with other released flits. Therefore, instead of unilaterally mapping tasks as far as possible for exploring the paths which have no communication contention, the maximum hop  $HPC_{max}$  will be restrictively set as the upper bound in our task mapping (Section 5), such that the data paths without conflict can be traversed straight from the source to the destination through bypass in a single cycle.

### 5 THERMAL-AWARE TASK MAPPING ON DYNAMICALLY TOPOLOGY-RECONFIGURABLE NOC

In this section, we propose a thermal-aware task mapping on dynamically topology-reconfigurable NoC in Algorithm 3, *TopoMap*. Unlike the conventional communication condensed task mapping methods, inter-processor distance seems non-critical for communication in our approach. Instead, communication conflict should be necessarily considered in the reconfiguration of SMART NoC as confirmed in Section 4. Therefore, in our approach, task mapping distance is restricted to be in the range of Lower Bound  $B_{Lower} = 2$  to Upper Bound  $B_{Upper} = HPC_{max}$ , which can greatly reduce the complexity of exploring all candidate cores in the design space, especially for large-scale NoCs with numerous processors.

As the detailed description in Algorithm 3, for a task graph composed of  $\mathcal{V}$  nodes,  $\mathcal{E}$  communication edges, it will be mapped on a NoC containing  $\mathcal{P}$  cores. In real situations, a fraction of cores (not restricted to 50%) will be activated, denoted as  $\mathcal{P}_A$ , which is restricted by chip safe temperature threshold  $T_{safe}$ . In any state, processors one-hop away from active cores will be powered-off as the examples shown in Fig. 13. When new tasks are released, dark cores with distance of 2 hops away from active cores will be chosen as available cores by calling Algorithm 1, **Temp\_Map** (line 10). Processors with the maximized minimum distance from all of currently active cores will be initially selected. Then, Algorithm 2, **Cont\_Map**, will be



**Algorithm 3** TopoMap

---

**Input:** 1) A set of cores  $\mathcal{P}$ ; and a set of active cores  $\mathcal{P}_A$ .  
 2) A set of task nodes  $\mathcal{V}$ ; and a set of edges  $\mathcal{E}, \mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ .  
**Output:** Task Mapping:  $\mathcal{M}_{v_i \rightarrow p_j}, v_i \in \mathcal{V}, p_j \in \mathcal{P}$ .

---

```

1:  $\mathcal{Q} \leftarrow$  Initial queue for  $v \in \mathcal{V}$  by breadth first search;
2: for  $v_i \in \mathcal{Q}$  do
3:   if  $\mathcal{P}_A = \mathcal{P}$  then
4:     Wait; /* All cores are active; chip temperature  $< \mathcal{T}_{safe}$  */
5:   end if
6:   for  $p_j \in (\mathcal{P} - \mathcal{P}_A)$  do
7:     for  $p_k \in \mathcal{P}_A$  do
8:        $Dis = |x_j - x_k| + |y_j - y_k|$ ;
9:       if  $B_{Lower} \leq Dis \leq B_{Upper}$  then
10:        /*  $B_{Lower} = 2$  and  $B_{Upper} = Dis \leq HPC_{max}$  */
11:         $Thermal\_map = Temp\_Map(v_i, p_j, \mathcal{P}_A)$ ;
12:         $Content\_map = Cont\_Map(v_i, p_j)$ ;
13:        if  $(Thermal\_map = 1) \wedge (Content\_map = 1)$  then
14:           $\mathcal{M}_{v_i \rightarrow p_j} = 1$ ; Break;
15:           $\mathcal{P}_A = \mathcal{P}_A - p_j$ ;
16:        end if
17:      end if
18:    end for
19:  end for
20: end for

```

---

called to explore communication contention-free routing paths for constructing communication topology among the candidate cores (line 11). If the selected cores can satisfy both of the requirements, it is the location for task mapping (line 12-14). Otherwise, dark cores with the distance of 3 hops to  $HPC_{max}$  hop away from the active cores will be checked, until the feasible solution is obtained. In the exploration of the available cores, cores with the closest distance will be checked first by the contention-aware and the temperature-aware approaches. If they cannot satisfy the constraint, the processors with one more hop will be checked in the same way, and then the cores with two more hops, until an available core is found. Even the processors which are finally selected are further with long distance, the inter communication can be conducted in a single cycle through bypass because there is no communication conflict due to the contention-aware task mapping.

In our work, we perform runtime task mapping and scheduling to improve application performance and system energy efficiency under safe temperature constraint, not limited by the constraint of Thermal Design Power (TDP) [46] or Thermal Safe Power (TSP) [47] budget. When an application is released, it will be mapped and executed on processors that will not violate safe chip temperature. The algorithm will be conducted on one of

the processing units, on which task mapping is not considered and energy consumption is calculated according to the power state of active core. We use *MatEx* [39] to obtain the transient temperature of all cores and calculate chip peak temperature at runtime in Algorithm 1, which can guarantee the validation of task mapping. In this paper, we mainly focus on the strategic design of the optimization algorithms and did not restrict the implementation to be in hardware or software.

The communication network topology will be reconfigured and the bypass will be built according to different task mappings and inter-processor flits transmissions. Thus, in our proposed approaches, by controlling the established long-distance communication paths, the topology of SMART NoC in terms of the inter-connectivity among cores is changed dynamically at runtime. The runtime reconfigurability of the SMART NoC is utilized by the proposed task mapping algorithms for the total performance optimization. To the best of our knowledge, system-level thermal-aware task mapping and scheduling, in which communication contention and inter-processor distance are considered, is firstly proposed to utilize the characteristics of SMART NoC in communication efficiency, and further control the reconfigurability of the communication network.

*TopoMap* is a polynomial-time algorithm, whose complexity is  $O(|\mathcal{N}| \cdot |\mathcal{P} - \mathcal{P}_A| \cdot |\mathcal{P}_A|)$ , where  $|\mathcal{N}|$  is the number of tasks,  $|\mathcal{P}_A|$  is the number of active cores, and  $|\mathcal{P} - \mathcal{P}_A|$  is the number of dark cores. The number of dark cores and the number of active cores are in the range of the total number of cores  $\mathcal{P}$ . Notice that  $HPC_{max}$  is a constant value as long as the SMART NoC architecture is designed. Therefore, the time complexity of the algorithm is  $O(|\mathcal{N}| \cdot |\mathcal{P}|^2)$ , which can be solved in polynomial time.

## 6 EVALUATION EXPERIMENT

In this section, we conduct sets of comprehensive experiments to evaluate the effectiveness of our thermal-aware task mapping in on-chip communication, application performance and system energy efficiency in dark silicon many-core systems. They are compared with state-of-the-art approaches on extensive PARSEC [37] benchmark suits, including *Blackscholes*, *Facesim*, *Vips* and *Swaptions*.

We base our experiment on  $N \times N$  Alpha 21264 core array in 22-nm technique NoCs ( $4 \times 4, 6 \times 6, 8 \times 8$  and

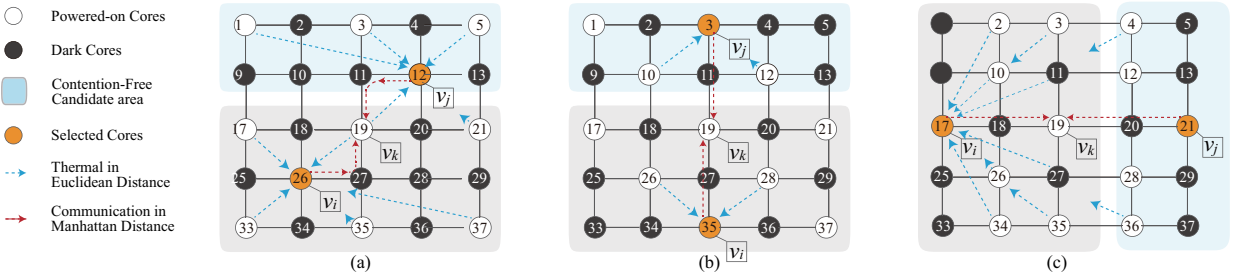


Fig. 13: Thermal-aware task mapping with the consideration of conflict under different  $HPC_{max}$ . (a)  $HPC_{max} = 2$ .  $p_{12}$  and  $p_{26}$  (or the alternative cores  $p_{10}$  and  $p_{28}$ ) can be selected since they are not out of the range of the maximum hop. (b)  $HPC_{max} = 3$ . Manhattan Distances from  $p_3, p_{35}$  to  $p_{19}$  are not larger than 3 (while  $MD(p_5, p_{19}) = 4$ ,  $MD(p_{33}, p_{19}) = 4$ ), meanwhile they have the maximized minimum distance from other active cores. (c)  $HPC_{max} = 4$ . Instead of other candidate processors within  $MD(p_{19}, p_i) = 4$  (e.g.,  $p_1, p_5$ ),  $p_{17}$  and  $p_{21}$  will be initially selected because  $MD(p_{17}, p_{19}) = 2$ ,  $MD(p_{21}, p_{19}) = 2$  are the shortest if  $\mathcal{T}_{safe}$  is maintained. Otherwise,  $p_9, p_{13}, p_{25}$  or  $p_{29}$  with 3 hops will be checked.

$10 \times 10$ ) as the target platforms and build the realistic platforms by GEM5 [48] simulator. GEM5 [48] has provided interpretation-based CPU models, on which we can build our target simulator for compute-system architecture with the encompassing system-level architecture as well as the processor microarchitectures and the communication realization. On top of it, the application mapping and scheduling can be accurately simulated as the processing in the real world system. Processor cores are homogeneous with the same execution efficiency, and are physically arranged as a dual-lane profile on the chip.

In the simulator, we model task mapping and scheduling with the simulated transmission latency. We assume the power of dark core is zero. Dynamic power and leakage power of an active core are obtained by *McPAT* model [33] in 22-nm technique. *McPAT* model [33] is accurate and widely used in power modeling, area, and timing simultaneously and consistently design space exploration for many-core systems. It can model the processor configuration based on the components of complete chip multiprocessor, network-on-chip, shared cache and memory control, such that the generated power model is accurate with real world operations. Steady state thermal maps of chip profile are generated by *HotSpot v 5.02* [34] with the configuration in grid model mode, which can exhibit chip thermal distribution. Runtime transient temperature and peak temperature are generated by the recently released *MatEx* [39] for run-time mapping and scheduling decisions, which can effectively compute the transient and peak temperature for the compact thermal models. Based on the matrix exponentials, it can be fast to calculate the temperatures from the input power traces for any given time resolutions without accuracy losses. Thus, the temperature will be computed only based on real-time running state of the system, not the previous temperature, such that the obtained results are accurate.

Parameters are detailed in Table 2, with an ambient temperature  $318.15K$  ( $45^\circ C$ ), and a threshold temperature  $354.95K$  ( $81.8^\circ C$ ) [35]. Experiments are run on a workstation with the Intel (R) E5-2650 at 2.6 GHz and 64 GB memory.

Experiments are carried out by comparing ILP model and heuristic algorithms with condensed task mapping approaches on mesh [16], NoC-Sprinting [28], and FoToNoC [25]. For the condensed task mapping approach on mesh-based NoC and FoToNoC, communication contention is not considered explicitly during task mapping. They take distant communication efficiency into account by using regular static topologies that facilitate the long-distance communication. In order to maintain the chip

thermal reliability [21][28], processor cores, even with long routing distance, will be activated for boosting the system performance [49]. While the communication efficiency and contention issues are not considered during their task mapping. In our presented ILP model and mapping algorithms, we have formally modeled network communication in space and time, and explored the mapping possibilities to reduce the contention occurrence, which will in turn increase the utilization of the single-cycle bypass in SMART NoC and improve the communication efficiency. We assume that the bandwidths of link channels are the same, and the latency of a flit transmission on a single link is one cycle.

## 6.1 Comparison of System Performance

We begin by evaluating the effectiveness of ILP model and algorithm *TopoMap* in communication and application performance. Applications are mapped on  $4 \times 4$ ,  $6 \times 6$  and  $8 \times 8$  NoCs with injection rate of applications (IRA) from 0.02 to 0.1 (generated by Negative Exponential Distribution [50]). We have also evaluated the performance of our presented approaches in the high saturated traffic workload which is highlighted in the x-coordinate. The maximum hop of bypass is  $HPC_{max} = 8$  [42]. In order to guarantee system thermal reliability, chip safe temperature  $T_{safe} = 81.8^\circ C$  is set as the constraint in task mapping and scheduling [35], on top of which we compare the inter-processor communication latency and the application schedule length in different approaches.

As experimental results shown in Fig. 14 and Fig. 15, on-chip communication latency and application performance regarding the schedule length are dramatically different from five approaches. ILP approach has guaranteed the best sequence of task execution and communication, such that it can always achieve the optimal solutions of communication efficiency as demonstrated in Fig. 14, while *TopoMap* can get near-optimal solutions compared with ILP. Results further show that it can improve the communication efficiency, where on average 46.8%, 41.5% and 39.3% reduction in inter-processor communication latency can be achieved, respectively, compared with that obtained by mesh NoC [16], NoC-Sprinting [28] and FoToNoC [25], due to the minimized communication contention and the maximized bypass routing. With the same task execution efficiency on homogeneous cores, the low-latency communication obtained by the *TopoMap* can result in averagely 32.7%, 33.5% and 31.1% of application performance improvement than other three approaches, respectively. We can observe that the results obtained in high saturated traffic load are consistent with that in lower IRAs. Specifically, with the increasing network traffic load, especially for the situations with high saturated workload, our presented ILP model and the heuristic algorithms can achieve more reduction in communication latency and improvement of performance-power efficiency than other techniques, since our approaches can effectively minimize communication contention and maximize the single-cycle multi-hop bypass routing. Results obtained by *TopoMap* are close to the optimal results

TABLE 2: Parameters in HotSpot [34], MatEx [39].

Parameter	Value
Thickness of the chip [mm]	0.15
Silicon thermal conductivity [W/m-K]	100
Silicon specific heat [ $J/m^3 \cdot K$ ]	$1.75 \times 10^6$
Heat sink convection capacitance [J/K]	140.4
Heat sink convection resistance [K/W]	0.1
Heat spreader thermal conductivity [W/m-K]	400
Heat sink spreader specific heat [ $J/m^3 \cdot K$ ]	$3.55 \times 10^6$
Interface material specific heat [ $J/m^3 \cdot K$ ]	$4 \times 10^6$
Ambient/Threshold temperature [K]	318.15 / 354.95



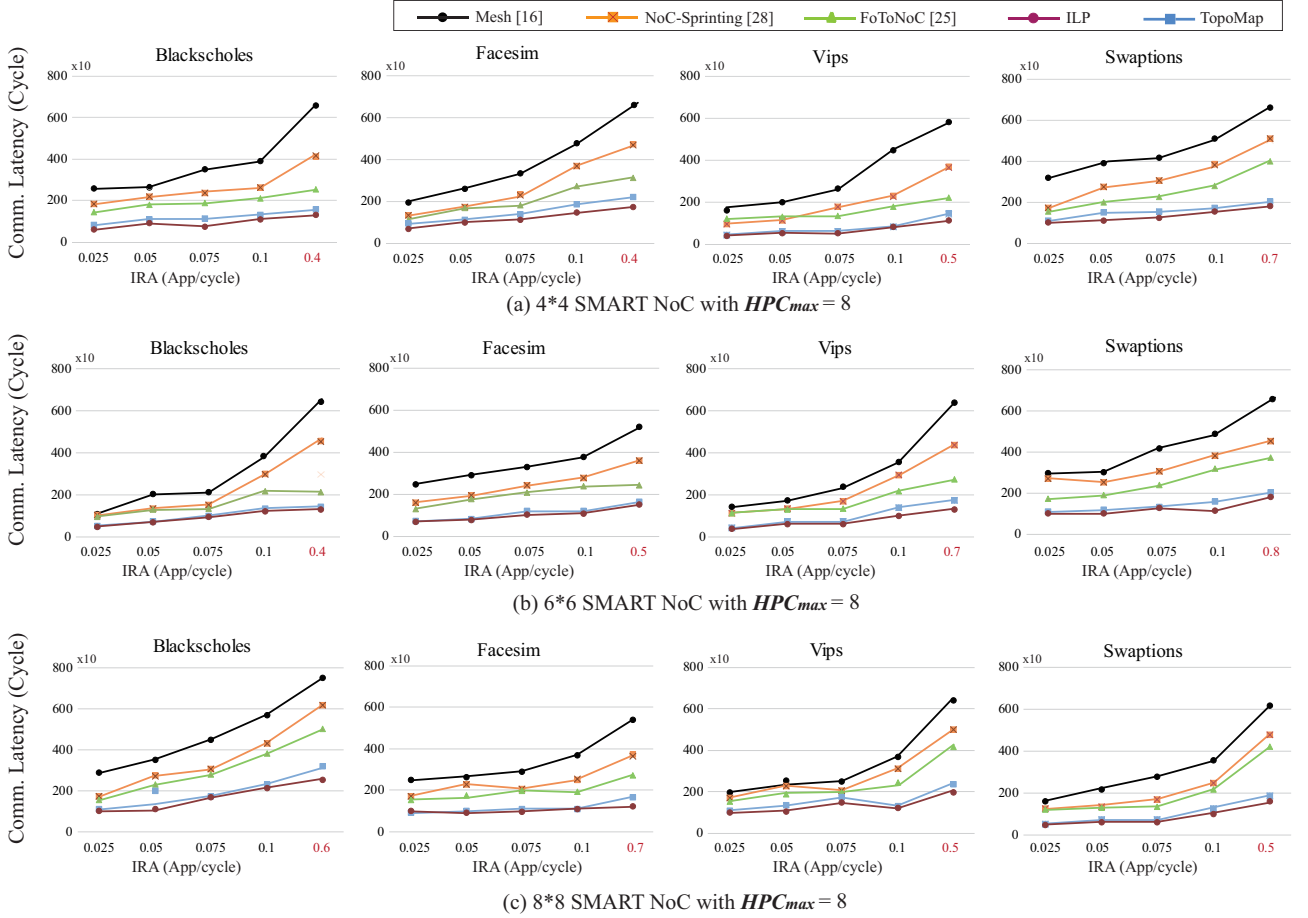


Fig. 14: Comparison results of communication latency of PARSEC benchmark applications on SMART NoCs.

obtained by the ILP approach, with a 1.8% difference on average as verified in Fig. 15.

There are several observations. First, with temperature constraint, cores are selectively activated on physical platform. It has lead to worse communication circumstances due to the long-distance path, since active cores are interconnected in a mesh topology and communicated via Hop-by-Hop message forwarding. Second, even cores are activated specifically on account of workload characteristics in [28], communication is proceeded as that in mesh. Third, physically distributed arrangement and logically adjacent interconnection of cores are beneficial to chip temperature and inter-communication to some extent due to the folded torus-like topology [25]. Applications mapped on cores in a cluster are non-centralized but communicated with the directly connected link, which has better performance than that in above two approaches. However, *FoToNoC* [25] is determined in design time, which is not feasible and will take unexpected overhead for long-distance communication. Fourth, *TopoMap* can explore the available thermal-reliable cores and dynamically reconfigure the communication topology between active cores via single-cycle bypass. It is much more accurate to approximate the complex runtime communication scenarios with a fine-grained consideration of communication contention. Communication latency can be greatly reduced by the maximized utilization of bypass for long-distance transmissions, such that application performance can be improved. Finally, *TopoMap* is efficient

in improving inter-processor communication, which has obtained the close results with the optimal solutions generated by ILP model. Note that, since we have considered real benchmark applications, the improvements can be slightly different (e.g. the application performance of four benchmarks on  $4 \times 4$  SMART NoC in Fig. 15) due to the criticality of the communication in the entire scheduling.

Then, we have conducted another set of experiments to evaluate the scalability of *TopoMap* with different  $HPC_{max}$  in four  $N \times N$  NoCs. We have recorded communication latency of a set of X264 benchmark applications with the increased  $HPC_{max}$  from 3 to 13. As shown in Fig. 16, evaluation results demonstrate that: i) Communication latency decreases rapidly when  $HPC_{max} \leq N$ . In this stage,  $HPC_{max}$  is the limitation of the efficiency enabled by bypass. Flits should be stopped and buffered in the  $HPC_{max}$  router if the inter-processor distance is longer than  $HPC_{max}$ . Thus, with the increase of  $HPC_{max}$ , more messages can be transmitted via only one bypass from the source to destination without buffering. Then, if  $HPC_{max} > N$  or even  $HPC_{max} \geq 2 \times (N - 1)$ , the communication overhead remains nearly constant since all cores are reachable via one bypass from any source routers. ii) With a same designed  $HPC_{max}$ , communication efficiency is better in larger scale NoCs. This can be easily understand that in large NoCs, there will be more available cores for exploring task mapping in which there is no communication contention. Without too much concern about the introduced distance between

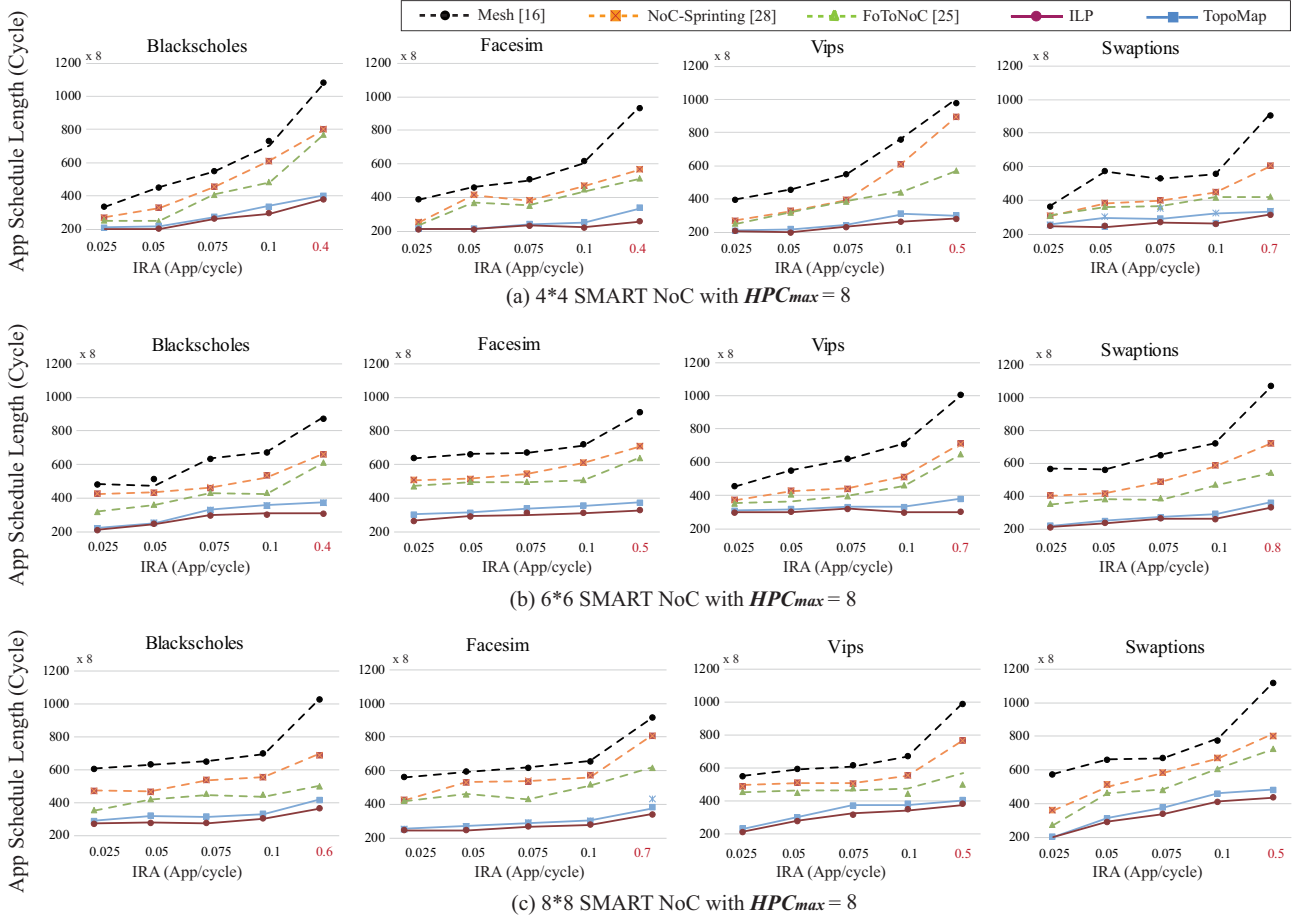


Fig. 15: Comparison results of application performance of PARSEC benchmark applications on SMART NoCs.

non-decentralized cores, messages can be forwarded via express bypass as long as there is no flits conflict. Thus, the utilization of bypass can be significantly increased and its efficiency can be enhanced in large-scale NoCs.

Third, we have recorded the running time of ILP, *TopoMap* and FoToNoC [25] in Table. 3. The resolving time is closely related to design space, that we have quantitatively analyzed the complexity of exploration space on account of the scales of applications and platforms. It is not surprising that the time complexity of ILP is exponentially increased, since the growth of the search space is exponential and ILP has introduced a new dimension for message priority assignment. It takes an average  $30\times$  more time to find the optimal solutions by ILP. We have integrated the ILP modeling skills to improve the efficiency. Nevertheless, it is worthwhile taking more efforts to explore the fine-grained communication optimization opportunities, not only due to the huge performance gain

that could be expected, but also because it is running at design-time for only once and then will probably be applied to millions of devices for runtime implementation. Meanwhile, *TopoMap* is polynomial-time algorithm, which is verified to be incomparably effective with the negligible running time ( $\leq 30$ ) compared with the time-costing ILP model. Besides, ILP and heuristic algorithms can respectively achieve at least 36.6% and 35.7% application performance than that by FoToNoC [25].

## 6.2 Comparison of Energy Consumption

Finally, we perform experiments to verify the effectiveness of *TopoMap* on system energy efficiency. For systematic energy consumption of many-cores fabricated as NoCs, it can be generally regarded as the total energy consumption of task execution  $E_e$  and communication  $E_c$ , denoted as  $E = E_e + E_c$ , where

$$\begin{aligned}
 E_e &= P \times t, \\
 E_c &= N \times \{ [E_{xbar} \times (h+1) + d \times E_{bf} \times (h+1) + \\
 &\quad E_{global} \times h + 2 \times E_{local}] \times L_{pk} + E_{cu} \times (h+1) \}.
 \end{aligned} \tag{13}$$

The computation energy increases linearly with the execution time since the per-core power is constant and obtained from power model *McPAT* [33]. Notice that the energy consumption of dark cores is zero. For the communication energy, the values of the constant parameters are obtained in 22-nm library by hardware

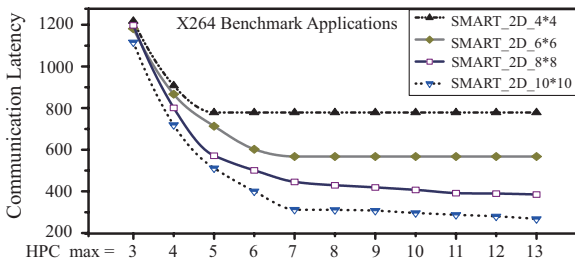


Fig. 16: Comparison of communication latency on NoCs with the increasing  $HPC_{max}$ .

TABLE 3: Design Space and Simulation Time of Benchmark Applications on 2-D SMART NoC Platforms.

Benchmarks		Design Space			Solutions									Simulation Time		
		4 × 4	6 × 6	8 × 8	4 × 4			6 × 6			8 × 8			Average		
					ILP	Topo	FoTo[25]	ILP	Topo	FoTo[25]	ILP	Topo	FoTo[25]	ILP	Topo	FoTo[25]
PARSEC	<i>Blackscholes</i>	1.8E+14	3.2E+16	7.9E+18	54	19	24	154	18	23	277	23	27	1mim	2s	2s
	<i>Facesim</i>	7.5E+22	2.5E+25	2.1E+28	57	13	27	266	23	31	461	20	29	3mims	4s	5s
	<i>Vips</i>	3.0E+26	4.3E+32	5.4E+39	193	15	31	453	16	25	661	21	31	1hr	7s	11s
Bench	<i>Swaption</i>	2.0E+24	3.1E+31	9.1E+36	251	21	24	457	19	27	765	28	39	5hrs	11s	16s
	<i>X264</i>	8.7E+30	2.7E+36	2.5E+41	351	24	27	758	21	29	1076	27	35	10hrs	18s	21s

synthesis in Hspice simulation [51]. The average unit energy to transmit a single bit through a crossbar is  $E_{xbar} = 0.108 \text{ pJ/bit}$ . It takes  $E_{global} = 0.031 \text{ pJ/bit}$  and  $E_{local} = 0.008 \text{ pJ/bit}$  to transmit a single bit through an electrical interconnect between routers, and through an electrical interconnect between a core and a router respectively. Averagely  $E_{bf} = 0.0078 \text{ pJ/bit}$  energy will be cost for buffering a bit, and  $E_{cu} = 0.917 \text{ pJ/packet}$  energy to make decisions for a packet.  $L_{pk} = 128 \text{ bit}$  is assumed as the packet size and  $d = 8$  is the buffer depth.  $N$  is the number of packets and  $h$  is the number of transmission hops obtained by simulation.

Evaluation results in Fig. 17 demonstrate the achievements on system energy consumption by our proposed ILP and heuristic approaches over other three state-of-the-art methods in different benchmark applications with the injection rate from 0.02 to 0.1. Since the energy consumption is determined by execution time when the execution configurations (e.g., frequency/voltage of the cores) are the same. As shown in Table. 3, the running time of our algorithm has been reduced on average 20.4% compared with others, which has demonstrated that the

energy consumption for executing our proposed algorithms will be much less than that of other approaches. Kindly note that when considering the entire application scheduling and the total system energy consumption, the running time of the algorithm is much less than the execution latency of benchmark applications as demonstrated in Table. 3. Therefore, to make the comparison results more focused, we only consider the energy consumption for executing benchmark applications on processors and the communication on network. We set the energy consumed by [16] as baseline for easy exhibition. Both of the proposed ILP approach and *TopoMap* method can significantly reduce system energy consumption than the state-of-the-art approaches due to the reduced overhead energy on transmission path. Besides, energy consumption of ILP can be further reduced over *TopoMap* since ILP approach can always find the optimal solutions in task mapping and scheduling, which will result in the minimized energy consumption. The processors in the lightest thermal region can be explored for contention-free task mapping, such that the lowest system energy consumption can be achieved due to the reduced energy

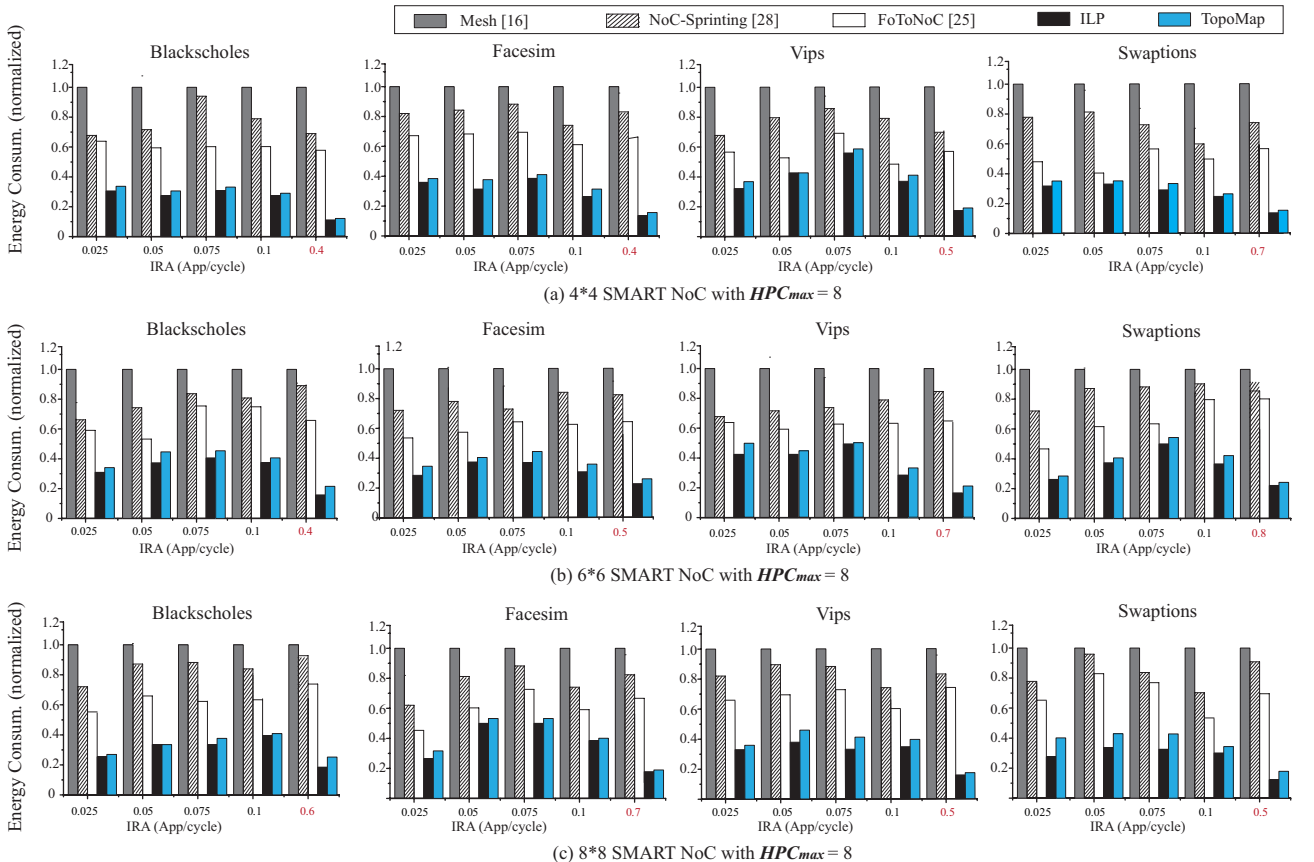


Fig. 17: Comparison results of system energy consumption of PARSEC benchmark applications on SMART NoCs.

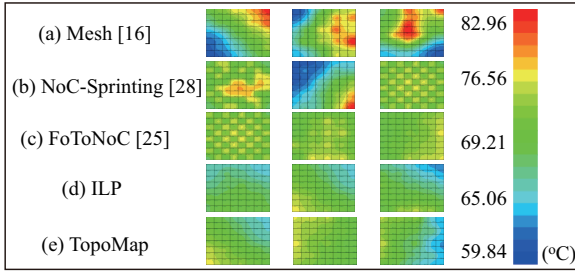


Fig. 18: Comparison of thermal distribution on X264.

consumed on inter-processor communication when the energy consumption on processors are the same with other approaches.

We can observe from the results that, with the increasing injection rate of applications, significant reductions in system energy consumption (on average 36.2%, 31.7%, 31.3% and 34.9%, 32.4%, 29.7%, respectively) can be achieved by the ILP approach and the *TopoMap* approach compared with that in mesh-based NoC [16], NoC-Sprinting [28] and FoToNoC [25]. Since the computation of tasks is equivalent in all systems with the implementation of homogeneous processors, the reduction in the total system energy consumption is beneficial from the reduced overhead of on-chip communication. The runtime topology-reconfigurable SMART NoC with single-cycle multi-hop bypass routing has dramatically decreased the energy consumption for arbitration, buffering and forwarding in intermediate routers and link channels, enabling the ideal message transmission from the source to destination router.

Finally, based on the resultant power traces, we can obtain steady state thermal profiles of heat distributions after running a set of *X264* benchmark applications. Partial thermal maps on mesh [16], NoC-Sprinting [28] and FoToNoC [25] platforms are shown in Fig. 18, compared with that obtained by ILP and *TopoMap*. Since we have set the safe temperature value as the threshold in the task mapping and scheduling, all of the executions are conducted under the constraint, which will not violate the thermal reliability. Nevertheless, as demonstrated in the figure, we can observe that the heat is much more concentrate on the other approaches comparing with that by ILP approach and *TopoMap*, since tasks are mapped on relatively distributed processors on SMART NoC.

Combining sets of experiments, results have verified the practicability and effectiveness of our proposed approaches. Distinct from conventional methods, our presented dynamically reconfigurable communication topology can be effectively controlled to guarantee safe chip temperature, meanwhile outperforming state-of-the-art static NoC designs in communication efficiency, application performance and system energy consumption for task mapping and scheduling in many-core systems.

## 7 CONCLUSION

In order to co-optimize chip thermal reliability and communication efficiency in dark silicon many-core systems, in this paper, we novelly presented an ILP model and a task mapping technique, *TopoMap*, on top of the reconfigurable SMART NoC architecture. NoC communi-

cation topology is dynamically reconfigured by thermal-aware and communication contention-aware task mapping, such that a thermal-safe system is guaranteed by physically decentralized active cores, and communication overhead is reduced by the minimized communication contention and the maximized bypass routing. Evaluation results on PARSEC benchmark applications have verified the applicability and effectiveness of the proposed techniques for improving communication efficiency, application performance, and reducing system energy consumption, compared with state-of-the-art techniques.

## ACKNOWLEDGMENT

This work is partially supported by NTU SUG and NAP, Singapore, NSFC 61772094, Chongqing High-Tech Program cstc2017jcyjA1430, Fundamental Research Funds for the Central Universities No. 106112017CDJQJ188829, China, and China Scholarship Council No. 201706050116 and No. 201706050117.

## REFERENCES

- [1] M. Shafique, S. Garg, T. Mitra, S. Parameswaran, and J. Henkel, "Dark silicon as a challenge for hardware/software co-design: Invited special session paper," in *Proceedings of the 2014 International Conference on Hardware/Software Codesign and System Synthesis*, p. 13, ACM, 2014.
- [2] C.-H. O. Chen, S. Park, T. Krishna, S. Subramanian, A. P. Chandrakasan, and L.-S. Peh, "Smart: a single-cycle reconfigurable noc for soc applications," in *Proceedings of the Conference on Design, Automation and Test in Europe*, pp. 338–343, EDA Consortium, 2013.
- [3] T. Krishna and L.-S. Peh, "Single-cycle collective communication over a shared network fabric," in *Networks-on-Chip (NoCS)*, 2014 Eighth IEEE/ACM International Symposium on, pp. 1–8, IEEE, 2014.
- [4] T. Krishna, C.-H. O. Chen, W.-C. Kwon, and L.-S. Peh, "Smart: single-cycle multihop traversals over a shared network on chip," *IEEE micro*, vol. 34, no. 3, pp. 43–56, 2014.
- [5] W. J. Dally, "Express cubes: Improving the performance of k-ary n-cube interconnection networks," *IEEE Transactions on Computers*, vol. 40, no. 9, pp. 1016–1023, 1991.
- [6] U. Y. Ogras and R. Marculescu, "it's a small world after all": Noc performance optimization via long-range link insertion," *IEEE Transactions on very large scale integration (VLSI) systems*, vol. 14, no. 7, pp. 693–706, 2006.
- [7] C. Jackson and S. J. Hollis, "Skip-links: A dynamically reconfiguring topology for energy-efficient nocs," in *System on Chip (SoC)*, 2010 International Symposium on, pp. 49–54, IEEE, 2010.
- [8] N. Bansal, S. Gupta, N. Dutt, A. Nicolau, and R. Gupta, "Network topology exploration of mesh-based coarse-grain reconfigurable architectures," in *Proceedings Design, Automation and Test in Europe Conference and Exhibition*, vol. 1, pp. 474–479, Vol.1, 2004.
- [9] E. J. Chang, H. K. Hsin, C. H. Chao, S. Y. Lin, and A. Y. Wu, "Regional aco-based cascaded adaptive routing for traffic balancing in mesh-based network-on-chip systems," *IEEE Transactions on Computers*, vol. 64, pp. 868–875, March 2015.
- [10] E. J. Chang, H. K. Hsin, S. Y. Lin, and A. Y. Wu, "Path-congestion-aware adaptive routing with a contention prediction scheme for network-on-chip systems," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 33, pp. 113–126, Jan 2014.
- [11] F. A. Samman, T. Hollstein, and M. Glesner, "Runtime contention and bandwidth-aware adaptive routing selection strategies for networks-on-chip," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, pp. 1411–1421, July 2013.
- [12] C. Li, D. Dong, and X. Liao, "Exploiting contention and congestion aware switch allocation in network-on-chips," in *Proceedings of the ACM Turing 50th Celebration Conference - China*, ACM TUR-C '17, (New York, NY, USA), pp. 41:1–41:10, ACM, 2017.
- [13] L. Yang, W. Liu, W. Jiang, W. Zhang, M. Li, J. Yi, D. Liu, and E. H. M. Sha, "Traffic-aware application mapping for network-on-chip based multiprocessor system-on-chip," in *2015 IEEE 17th International Conference on High Performance Computing and Communications*, pp. 571–576, Aug 2015.
- [14] A. Davare, J. Chong, Q. Zhu, D. M. Densmore, and A. L. Sangiovanni-Vincentelli, "Classification, customization, and characterization: Using milp for task allocation and scheduling," *Systems Research*, 2006.
- [15] N. Koziris, M. Romesis, P. Tzanakas, and G. Papakonstantinou, "An efficient algorithm for the physical mapping of clustered task graphs onto multiprocessor architectures," in *Parallel and Distributed Processing, 2000. Proceedings. 8th EuroMicro Workshop on*, pp. 406–413, IEEE, 2000.
- [16] H. Yu, Y. Ha, and B. Veeravalli, "Communication-aware application mapping and scheduling for noc-based mpocs," in *Circuits and Systems (ISCAS)*, Proceedings of 2010 IEEE International Symposium on, pp. 3232–3235, IEEE, 2010.
- [17] E. Carvalho, N. Calazans, and F. Moraes, "Heuristics for dynamic task mapping in noc-based heterogeneous mpocs," in *18th IEEE/IFIP International Workshop on Rapid System Prototyping (RSP '07)*, pp. 34–40, May 2007.
- [18] W. Jiang, E. H.-M. Sha, Q. Zhuge, and X. Chen, "Optimal functional-unit assignment and buffer placement for probabilistic pipelines," in *Hardware/Software Codesign and System Synthesis (CODES+ISSS)*, 2016 International Conference on, pp. 1–10, IEEE, 2016.
- [19] W. Jiang, E. H.-M. Sha, Q. Zhuge, L. Yang, H. Dong, and X. Chen, "On the design of minimal-cost pipeline systems satisfying hard/soft real-time constraints," *IEEE Transactions on Emerging Topics in Computing*, 2018.
- [20] A. K. Singh, M. Shafique, A. Kumar, and J. Henkel, "Mapping on multi-/many-core systems: Survey of current and emerging trends," in *Proceedings of the 50th Annual Design Automation Conference, DAC '13*, (New York, NY, USA), pp. 1:1–1:10, ACM, 2013.
- [21] H. Khdr, S. Pagani, M. Shafique, and J. Henkel, "Thermal constrained resource management for mixed ilp-tlp workloads in dark silicon chips," in *Proceedings of the 52nd Annual Design Automation Conference, DAC '15*, (New York, NY, USA), pp. 179:1–179:6, ACM, 2015.
- [22] L. Yang, W. Liu, W. Jiang, M. Li, J. Yi, and E. H. M. Sha, "Application mapping and scheduling for network-on-chip-based multiprocessor system-on-chip with fine-grain communication optimization," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 24, pp. 3027–3040, Oct 2016.
- [23] J. Wang, Z. Chen, J. Guo, Y. Li, and Z. Lu, "Aco-based thermal-aware thread-to-core mapping for dark-silicon-constrained cmcs," *IEEE Transactions on Electron Devices*, vol. 64, pp. 930–937, March 2017.
- [24] H. Khdr, S. Pagani, V. Sousa, V. Lari, A. Pathania, F. Hannig, M. Shafique, J. Teich, and J. Henkel, "Power density-aware resource management for heterogeneous tiled multicores," *IEEE Transactions on Computers*, vol. 66, pp. 488–501, March 2017.
- [25] L. Yang, W. Liu, W. Jiang, M. Li, J. Yi, and E. H.-M. Sha, "Fotonoc: A hierarchical management strategy based on folded lorus-like network-on-chip for dark silicon many-core systems," in *Design Automation Conference (ASP-DAC)*, 2016 21st Asia and South Pacific, pp. 725–730, IEEE, 2016.



- [26] L. Yang, W. Liu, W. Jiang, M. Li, P. Chen, and E. H. M. Sha, "Fotonoc: A folded torus-like network-on-chip based many-core systems-on-chip in the dark silicon era," *IEEE Transactions on Parallel and Distributed Systems*, vol. 28, pp. 1905–1918, July 2017.
- [27] L. Yang, W. Liu, W. Jiang, C. Chen, M. Li, P. Chen, and E. H. Sha, "Hardware-software collaboration for dark silicon heterogeneous many-core systems," *Future Generation Computer Systems*, vol. 68, pp. 234–247, 2017.
- [28] J. Zhan, Y. Xie, and G. Sun, "Noc-sprinting: Interconnect for fine-grained sprinting in the dark silicon era," in *Design Automation Conference (DAC)*, 2014 51st ACM/EDAC/IEEE, pp. 1–6, IEEE, 2014.
- [29] H. Bokhari, H. Javadi, M. Shafique, J. Henkel, and S. Parameswaran, "darknoc: Designing energy-efficient network-on-chip with multi-vt cells for dark silicon," in *Proceedings of the 51st Annual Design Automation Conference*, pp. 1–6, ACM, 2014.
- [30] M. Shafique, S. Garg, J. Henkel, and D. Marculescu, "The eda challenges in the dark silicon era: Temperature, reliability, and variability perspectives," in *Proceedings of the 51st Annual Design Automation Conference*, pp. 1–6, ACM, 2014.
- [31] D. Kline Jr, K. Wang, R. Melhem, and A. K. Jones, "Mscs: Multi-hop segmented circuit switching," in *Proceedings of the 25th annual conference on Great Lakes Symposium on VLSI*, pp. 179–184, ACM, 2015.
- [32] L. Yang, W. Liu, N. Guan, M. Li, P. Chen, and H. Edwin, "Dark silicon-aware hardware-software collaborated design for heterogeneous many-core systems," in *Design Automation Conference (ASP-DAC)*, 2017 22nd Asia and South Pacific, pp. 494–499, IEEE, 2017.
- [33] S. Li, J. H. Ahn, R. D. Strong, J. B. Brockman, D. M. Tullsen, and N. P. Jouppi, "Mcpat: an integrated power, area, and timing modeling framework for multicore and manycore architectures," in *Microarchitecture, 2009. MICRO-42. 42nd Annual IEEE/ACM International Symposium on*, pp. 469–480, IEEE, 2009.
- [34] W. Huang, S. Ghosh, S. Velusamy, K. Sankaranarayanan, K. Skadron, and M. R. Stan, "Hotspot: A compact thermal modeling methodology for early-stage vlsi design," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 14, no. 5, pp. 501–513, 2006.
- [35] K. Skadron, M. R. Stan, W. Huang, S. Velusamy, K. Sankaranarayanan, and D. Tarjan, "Temperature-aware microarchitecture," in *Computer Architecture, 2003. Proceedings. 30th Annual International Symposium on*, pp. 2–13, IEEE, 2003.
- [36] M. Fattah, M. Daneshmand, P. Liljeberg, and J. Plosila, "Smart hill climbing for agile dynamic mapping in many-core systems," in *Proceedings of the 50th Annual Design Automation Conference*, p. 39, ACM, 2013.
- [37] C. Bienia, S. Kumar, J. P. Singh, and K. Li, "The parsec benchmark suite: Characterization and architectural implications," in *Proceedings of the 17th international conference on Parallel architectures and compilation techniques*, pp. 72–81, ACM, 2008.
- [38] T. Chantem, X. S. Hu, and R. P. Dick, "Temperature-aware scheduling and assignment for hard real-time applications on mpsoes," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 19, no. 10, pp. 1884–1897, 2011.
- [39] S. Pagani, J.-J. Chen, M. Shafique, and J. Henkel, "Matex: Efficient transient and peak temperature computation for compact thermal models," in *Proceedings of the 2015 Design, Automation & Test in Europe Conference & Exhibition*, pp. 1515–1520, EDA Consortium, 2015.
- [40] T.-S. Hsu, J.-L. Chiu, C.-K. Yu, and J.-J. Liou, "A fast and accurate network-on-chip timing simulator with a flit propagation model," in *Design Automation Conference (ASP-DAC)*, 2015 20th Asia and South Pacific, pp. 797–802, IEEE, 2015.
- [41] P. Lotfi-Kamran, M. Modarressi, and H. Sarbazi-Azad, "An efficient hybrid-switched network-on-chip for chip multiprocessors," *IEEE Transactions on Computers*, vol. 65, no. 5, pp. 1656–1662, 2016.
- [42] T. Krishna, C.-H. O. Chen, W. C. Kwon, and L.-S. Peh, "Breaking the on-chip latency barrier using smart," in *High Performance Computer Architecture (HPCA2013)*, 2013 IEEE 19th International Symposium on, pp. 378–389, IEEE, 2013.
- [43] C.-K. Zhang, Y. Cui, H.-Y. TANG, and J.-P. Wu, "State-of-the-art survey on software-defined networking (sdn)," *Journal of software*, vol. 26, no. 1, pp. 62–81, 2015.
- [44] W. Jiang, E. H.-M. Sha, X. Chen, L. Yang, L. Zhou, and Q. Zhuge, "Optimal functional-unit assignment for heterogeneous systems under timing constraint," *IEEE Transactions on Parallel and Distributed Systems*, vol. 28, no. 9, pp. 2567–2580, 2017.
- [45] V. Zivojnovic, "Dsp stone: A dsp-oriented benchmarking methodology," *Proc. of ICSPAT*, 1994, 1994.
- [46] A. Pathania, H. Khdr, M. Shafique, T. Mitra, and J. Henkel, "Scalable probabilistic power budgeting for many-cores," in *Design, Automation Test in Europe Conference Exhibition (DATE)*, 2017, pp. 864–869, March 2017.
- [47] S. Pagani, H. Khdr, J.-J. Chen, M. Shafique, M. Li, and J. Henkel, *Thermal Safe Power: Efficient Thermal-Aware Power Budgeting for Manycore Systems in Dark Silicon*, pp. 125–158. Cham: Springer International Publishing, 2017.
- [48] N. Binkert, B. Beckmann, G. Black, S. K. Reinhardt, A. Saidi, A. Basu, J. Hestness, D. R. Hower, T. Krishna, S. Sardashti, et al., "The gem5 simulator," *ACM SIGARCH Computer Architecture News*, vol. 39, no. 2, pp. 1–7, 2011.
- [49] A. K. Singh, M. Shafique, A. Kumar, and J. Henkel, "Resource and throughput aware execution trace analysis for efficient run-time mapping on mpsoes," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 35, pp. 72–85, Jan 2016.
- [50] A.-M. Rahmani, I. Kamali, P. Lotfi-Kamran, A. Afzali-Kusha, and S. Safari, "Negative exponential distribution traffic pattern for power/performance analysis of network on chips," in *VLSI Design, 2009 22nd International Conference on*, pp. 157–162, IEEE, 2009.
- [51] W. Liu, W. Zhang, X. Wang, and J. Xu, "Distributed sensor network-on-chip for performance optimization of soft-error-tolerant multiprocessor system-on-chip," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 24, no. 4, pp. 1546–1559, 2016.



**Weichen Liu** (S'07-M'11) is an assistant professor at School of Computer Science and Engineering, Nanyang Technological University, Singapore. He received the PhD degree from the Hong Kong University of Science and Technology, Hong Kong, and the BEng and MEng degrees from Harbin Institute of Technology, China. Dr. Liu has more than 70 research papers and received Best Paper Candidate Awards from CODES+ISSS, CASES and ASP-DAC. His research interests

include embedded and real-time systems, multiprocessor systems and fault-tolerant systems.



**Lei Yang** received the B.E. degree from the School of Computer Science at Chongqing University, Chongqing, China, in 2013 and is currently pursuing the Ph.D degree from the Department of Computer Science at Chongqing University, Chongqing, China. Concurrently, she is visiting in the Department of Information and Computer Science at the University of California, Irvine. Her current research interests include NoC optimization, multi-core architecture design and high-performance computing in Nonvolatile Memory (NVM) based NoCs.



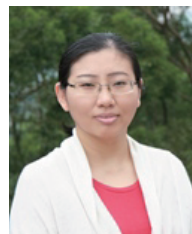
**Weiwen Jiang** received the B.E. degree from the Department of Computer Science, Nanjing Agricultural University, Nanjing, China, in 2012. He is currently working towards the Ph.D degree in the Department of Computer Science at Chongqing University. Concurrently, he is a visiting scholar in University of Pittsburgh. His current research interests include self-timed system optimization, embedded systems, and NVM.



**Liang Feng** received the PhD degree from School of Computer Engineering, Nanyang Technological University, Singapore, in 2014. He was a Postdoctoral Research Fellow at the Computational Intelligence Graduate Lab, Nanyang Technological University, Singapore. He is currently an Assistant Professor at the College of Computer Science, Chongqing University, China. His research interests include Computational and Artificial Intelligence and Transfer Learning.



**Nan Guan** is currently an assistant professor at the Department of Computing, The Hong Kong Polytechnic University. Dr Guan received his BE and MS from Northeastern University, China in 2003 and 2006 respectively, and a PhD from Uppsala University, Sweden in 2013. Before joining PolyU in 2015, he worked as a faculty member in Northeastern University, China. His research interests include real-time embedded systems and cyber-physical systems. He received the EDAA Outstanding Dissertation Award in 2014, the Best Paper Award of RTSS 2009 and DATE 2013.



**Wei Zhang** received the B.S and M.S degrees from Harbin Institute of Technology, Harbin, China, in 1999 and 2001, respectively, and the Ph.D. degree from Princeton University, Princeton, NJ, USA in 2009. She is currently an Associate Professor with the Department of Electronic and Computer Engineering, the Hong Kong University of Science and Technology, Hong Kong. She was an Assistant Professor with the School of Computer Engineering, Nanyang Technological University, Singapore, from 2010 to 2013. Her current research interests include reconfigurable system, power and thermal management, embedded system security, and emerging technologies. She currently serves as an Area Editor of Reconfigurable Computing of ACM TECS, Associate Editor of IEEE TVLSI, and ACM JETC, and serves on conference organization and technical program committees.



**Nikil Dutt** received a Ph.D. in Computer Science from the University of Illinois at Urbana-Champaign in 1989, and is currently a Chancellor's Professor at the University of California, Irvine, with academic appointments in the CS, EECS, and Cognitive Sciences departments. His research interests are in embedded systems, EDA, computer systems architecture and software, and brain-inspired architectures and computing. He received numerous best paper awards at conferences

Dutt previously served as Editor-in-Chief of ACM TODAES and as Associate Editor for ACM TECS and IEEE TVLSI. He has served on several premier EDA and Embedded System Design conferences and workshops, and serves or has served on the advisory boards of ACM SIGBED, ACM SIGDA, ACM TECS and IEEE Embedded Systems Letters (ESL). He is a Fellow of the ACM, Fellow of the IEEE, and recipient of the IFIP Silver Core Award.