A Flexible Energy- and Reliability-Aware Application Mapping for NoC-Based Reconfigurable Architectures

Leibo Liu, Chen Wu, Chenchen Deng, Shouyi Yin, Qinghua Wu, Jie Han, and Shaojun Wei

***Abstract*— This paper proposes a flexible energy- and reliability-aware application mapping approach for network-on-chip (NoC)-based reconfigurable architecture. A parameterized cost model is first developed by combining energy and reliability with a weight parameter that defines the optimization priority. Using this model, the overall mapping cost could be evaluated. Subsequently, a mapping method using branch and bound with a partial cost ratio is employed to find the best mapping by enumerating all the possible patterns organized in a search tree. To improve the search efficiency, nonoptimal mappings are discarded at early stages using the partial cost ratio. Using the proposed approach, applications can be mapped onto most NoC topologies and running with various routing algorithms when considering both energy and reliability. Other state-of-the-art works have also done substantial research for the same topic but only limited to a specific topology or routing algorithm. Even for the same topology and routing algorithm, the proposed approach still shows considerable advantages in many aspects. Experiments show that this approach gains not only significant reduction in energy but also improvement in reliability. It also outperforms other approaches in throughput and latency with competitive run time.**

***Index Terms*— Energy, mapping algorithm, network on chip (NoC), reconfigurable hardware, reliability.**

I. Introduction

EQUIPPED with the flexibility of general-purpose processors (GPPs) and the efficiency of application-specific integrated circuits, reconfigurable architectures have proven their advantages in tremendous application areas [1], [2]. As a promising communication infrastructure, the network on chip (NoC) has a significant impact on the energy consumption and reliability of reconfigurable systems. Previous work has shown that the energy consumption of an NoC is more than 28% of the total system energy [3], [4], so it must be considered seriously. As critical for system interconnections, the reliability of an NoC is also of great importance. For example, a super parallel computer requires that its interconnection operates without a packet loss for 10 000 h [5]. However, it is difficult for the NoC to remain reliable due to the effects of various faults including alpha and neutron particle strikes [6], electromagnetic interference [7], power supply disturbances [8], and crosstalk [9]. Moreover, some real GPPs, such as the future payload data processing cores for science, earth, and telecommunication missions identified by the European and American space agencies [10]–[12], require the communication infrastructure to be both low power consumption and high reliability by fault tolerance. Hence, it becomes important to optimize both the energy and reliability of the NoC in a reconfigurable system during an application mapping.

While it is already challenging to meet both energy and reliability requirements for a specific topology and routing algorithm, it is mandatory for an application mapping in reconfigurable architectures to be applicable to diverse topologies and routing algorithms. This is because the flexibility of the reconfigurable architecture is achieved by dynamically changing the configuration of processing element (PE) functions and interconnections. Thus, the topologies and routing algorithms for NoC are different from one another in various applications scenarios. Consequently, a highly flexible application mapping is desired to meet the requirement of a reconfigurable system. In addition, run-time reconfiguration is also crucial for a reconfigurable architecture because faults must be avoided within a very short time during dynamic reconfiguration. This means that the computational overhead of an application mapping needs to be minimized and thus, the speed of the application mapping is of great importance as well.

Much effort has recently been made to optimize the energy consumption during a mapping process [13]–[16]. Work has also been done for maximizing the reliability of various mapping patterns [17]–[19]. In addition, optimizations have been considered for both energy and reliability [6], [20], [21]. In [6], a fault-tolerant mapping technique is proposed for optimizing both system performance and communication energy. A best mapping pattern, which achieves the highest reliability and minimal energy, for multiple applications has been obtained in [20]. In [21], a quantitative model of energy and reliability is proposed for finding the best mapping pattern. Although energy reduction and reliability improvement can be

Manuscript received November 15, 2013; revised April 15, 2014 and August 9, 2014; accepted October 23, 2014. This work was supported by the National High Technologies Research Program of China under Grant 2012AA012701.

L. Liu, C. Wu, C. Deng, S. Yin, Q. Wu, and S. Wei are with the National Laboratory for Information Science and Technology, Institute of Microelectronics, Tsinghua University, Beijing 100084, China (e-mail: liulb@tsinghua.edu.cn; wuchen12@mails.tsinghua.edu.cn; chenchendeng@tsinghua.edu.cn; yinsy@tsinghua.edu.cn; wuqinghuabest@126.com; wsj@tsinghua.edu.cn).

J. Han is with the Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB T6G 2V4, Canada (e-mail: jhan8@ualberta.ca).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVLSI.2014.2367108

obtained by these approaches, a common disadvantage is the low flexibility that limits their applications to a specific NoC topology and/or a routing algorithm. This indicates that these application mappings are not best suited for reconfigurable systems.

To provide an optimum application mapping method for NoC-based reconfigurable systems, two major contributions are made in this paper.

1) A parameterized cost model (PCM) that combines energy and reliability is developed. Most importantly, PCM is independent on the topologies of the NoC or the routing algorithms. The model is the fundamental factor that guarantees the flexibility of the mapping approach and the accuracy of the metrics, which include both energy consumption and reliability. In other words, the cost model is both energy and reliability aware and flexible. A performance constraints model is utilized to reduce the overheads in terms of throughput and latency. By integrating the performance constraints into PCM, it provides a flexible and accurate cost model for the mapping approach to select the best candidates.
2) A mapping approach called branch and bound with partial cost ratio (BBPCR) is proposed to figure out the best mapping pattern using PCM with a low computational overhead. The computation complexity is reduced significantly, but the accuracy of the search results is not sacrificed. Substantial experiments are performed to compare this paper with state-of-the-art approaches and quasi-theoretical boundary. Results not only show the high flexibility of the proposed approach, but also show the advantages of BBPCR in terms of energy, reliability, throughput, latency, and run time.

The rest of this paper is organized as follows. Section II introduces design considerations. In Section III, both the cost model PCM and the performance model are analyzed. Section IV discusses the mapping approach BBPCR in detail. Experimental results are presented in Section V. Section VI concludes this paper and discusses future work.

II. Design Consideration

The primary objective of this paper is to propose a highly flexible mapping approach on the premise that it will not incur extra overheads. This method can be applied to NoC-based reconfigurable architecture, which requires the flexibility of the mapping approach to be fit for various NoC topologies and routing algorithms. Meanwhile, the best mapping pattern is expected to have an optimal tradeoff between energy, reliability, and performance, or at least it does not incur significant extra cost. In addition, the dynamic reconfiguration requires a configuration to be implemented in runtime. This brings up a time limit for finding the best mapping pattern. In other words, the computation complexity of the approach needs to be carefully controlled.

However, it is difficult to simultaneously optimize several metrics such as energy, reliability, and performance. When faults occur in a system, reliability can be enhanced by bypassing the faulty components. However, it adds more components on the new communication path that leads to a larger energy consumption. Meanwhile, longer communication path may also cause increased latency and reduced throughput. The conflict between flexibility and computation complexity is an even more difficult challenge. A general model without utilizing the characteristics of some special topologies and routing algorithms can result in an increase in the computational complexity. Therefore, it is very challenging to obtain an acceptable tradeoff between these metrics.

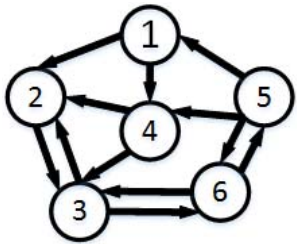


Fig. 1. One example of the topology of NoC.

To simultaneously optimize energy and reliability, a unified model is needed to measure energy and reliability quantitatively. Moreover, the unified model is required to be general for most topologies and routing algorithms. Since the performance evaluation is essential in the process of application mapping, a performance constraint model is required to estimate the throughput and latency of the system. When designing the mapping approach, the first question is how to accurately find the best mapping pattern based on these two models. Subsequently, how to efficiently choose the best pattern among a large number of candidate mapping patterns is also important to be considered.

III. Model Analysis

In this section, the backgrounds of the application and NoC architectures are first outlined; then, a unified cost model and a performance model are introduced.

The application used in this paper is presented as an application characteristic graph (APCG) $G(C, A)$ [22]. $G(C, A)$ is a directed graph, where each vertex $c_i \in C$ represents an intellectual property (IP) core, each edge $a_{ij} \in A$ represents the communication between source core c_i and destination core c_j, and the weight of each edge V^{ij} represents the communication volume in bits from source core c_i to destination core c_j. An example for the topology considered in this paper is shown in Fig. 1. Vertices and edges of the graph stand for the nodes and links of the NoC, respectively. Each node is composed of a PE and a router. Every PE performs its own computation and storage or input/output (I/O) processing functionality, and it communicates with the routers that are responsible for forwarding data packets to other routers. Links connect the routers and for a specific topology, the total number of links is defined as L, e.g., for the topology shown in Fig. 1, $L = 12$.

The reliability of an NoC can be affected by the errors in PEs, routers, or links. When PEs are faulty, spare PEs are often used to replace the faulty ones [6], [20], [23] so that the faults can be tolerated. In this way, the mapping is altered to

work on a different topology and routing algorithm. Therefore, PEs can be assumed to be fault free in this paper. As routers are connected to each other by the links around them, the faults in routers can be classified into the faults that occur in links around them [23]. In this way, only faults in links are considered, and they consist of the raw errors in links and the classified errors in routers.

Although links can have permanent faults and transient faults, only transient faults (soft errors) in links are considered in this paper. Permanent fabrication errors are not in the scope of discussion. Transient errors in links have been modeled to evaluate the reliability of NoC in [24]–[26], but the probability of faulty links are assumed to be the same, which is not true in the real world. Therefore, in the design of the proposed cost model, all the links can have different probabilities being faulty. When $l(l < L)$ links are reported to be faulty, we have $M = \binom{L}{l}$ different conditions considering the different positions of the l faulty links. All M conditions are considered when evaluating the energy and reliability, which will be discussed in detail in the following sections.

A. Unified Cost Model

In different application mapping scenarios, the requirement for energy and reliability varies. In some cases, the minimum energy consumption is desired, while in some other cases, the priority is to obtain the maximum reliability. Hence, an effective cost model must differentiate the significance of energy and reliability. In fact, as discussed above, reliability enhancement is often obtained at a cost of increased energy consumption. Thus, a weight parameter $\alpha \in [0, 1]$ is introduced. This parameter gives the weight between energy and reliability for different application requirements. With this parameter, a unified model is developed for taking one of energy and reliability as the major consideration in a joint optimization. Since the energy consumption is desired to be as low as possible while the higher the reliability the better a system is, *reliability* is replaced by *reliability cost* in this model. Therefore, the overall cost can be evaluated using this unified model as follows:

$$\text{Cost} = (1 - \alpha)EN + \alpha RN \tag{1}$$

where EN and RN are the normalized energy and reliability cost, respectively. The measurements of energy and reliability cost are described as follows.

1) Measurement of Communication Energy: The energy consumption usually consists of the static energy and the dynamic energy. The static energy is mainly influenced by the working temperature, the level of process technology, and the gate–source/drain–source voltages. In this paper, energy model is used for evaluating different mapping patterns for a certain application to be mapped onto the same NoC. This means that the temperature and process technology are all the same for different patterns. The variation of gate–source/drain–source voltages is quite small and therefore, the static energy is not considered in the proposed model. On the contrary, it is the dynamic energy (i.e., communication energy) that different mapping patterns have considerable impact on. As a result, the communication energy is the main concern to evaluate a certain mapping pattern. The energy in (1) is considered to be communication energy consumption and in the following discussion *energy* is referred to *communication energy*. The bit energy metric in [27] that gives the energy consumed by 1 bit of data transporting through a link and a router is used to model the communication energy. Consider the ith condition of all M possibilities when l links are defective. We use d_i^{SD} to represent the number of links that the data go through from the source node S to the destination node D. Then the energy consumed by communications between S and D can be calculated by

$$E_i^{\text{SD}} = V_i^{\text{SD}}\left(E_{L\text{bit}}d_i^{\text{SD}} + E_{R\text{bit}}\left(d_i^{\text{SD}} + 1\right)\right) \tag{2}$$

where V_i^{SD} is the communication volume in bits from source node S to destination node D, and $E_{L\text{bit}}$ and $E_{R\text{bit}}$ are the energy consumed by 1 bit of data transporting through a link and a route, respectively. The energy consumed by a specific mapping pattern of the network when l links are faulty can be expressed by

$$E_i^l = \sum_{\text{SD}} V_i^{\text{SD}}\left(E_{L\text{bit}}d_i^{\text{SD}} + E_{R\text{bit}}\left(d_i^{\text{SD}} + 1\right)\right)P_i^{\text{SD}} \tag{3}$$

where P_i^{SD} is set to be 1 when there is a communication between S and D; otherwise, it is set to be 0, i.e., as expressed in

$$P_i^{\text{SD}} = \begin{cases} 1, & a_{\text{SD}} \in A \\ 0, & a_{\text{SD}} \notin A \end{cases} \tag{4}$$

where a_{SD} represents the communication between S and D and A indicates the collection of all the communication of the APCG. Equation (3) shows that E_i^l changes with i or l as once i or l changes and the communication path between source and destination becomes different, which leads to a change in energy consumption. Therefore, to accurately evaluate the energy consumption, E_i^l is regarded as a discrete variable on the number of faulty links l and condition i. The total energy consumption by an NoC under a specific mapping pattern is expressed as (5). The model can give accurate energy estimation close to the reality because all the faulty scenarios are taken into consideration

$$E = \sum_{l=0}^{L}\sum_{i=1}^{M} E_i^l P_i \tag{5}$$

where $P_i = \prod_{j=0}^{l} p_j \times \prod_{j=0}^{L-l}(1 - p_j)$ is the probability that l links are faulty, and p_j is the corresponding faulty probability of each link that is independent of each other. Subsequently, the absolute value of E in (5) is normalized with a normalization factor EN', which is obtained by considering the worst case of d_i^{SD} being the maximum

$$EN = E/EN'. \tag{6}$$

2) Measurement of Reliability Cost: When considering the reliability problem of NoC, error codec is very efficient to reduce or even eliminate the soft errors in NoC. Instead of rectifying the errors, the reliability model proposed in this paper is used to evaluate and qualify the reliability of all the

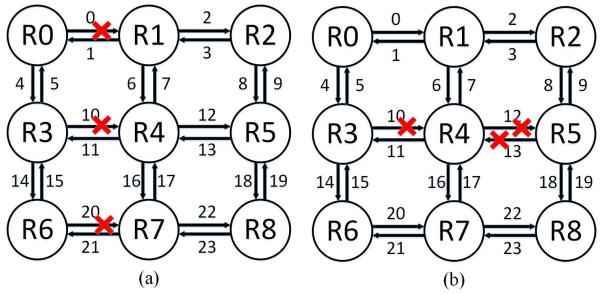


Fig. 2. Two fault patterns with three faulty links. (a) Links numbered 0, 10 and 20 are faulty. (b) Links numbered 10, 12 and 13 are faulty.

possible mapping patterns when soft errors exist. In this way, the reliability model helps the mapping approach select the best mapping candidate by providing accurate reliability cost. The relationship between the reliability cost and reliability is that the higher the reliability cost is, the lower the reliability is. The reliability cost of a communication pair is defined to be zero if the pair can tolerate all the defective links; otherwise, it is one. For example, if the links 0, 10, and 20 are faulty in Fig. 2(a), no path from R3 to R4 can tolerate all the faults for transporting data. In this case, the reliability cost is considered to be one. However, if the links 10, 12 and 13 are faulty, as shown in Fig. 2(b), the path R3$\longrightarrow$R0$\longrightarrow$R1$\longrightarrow$R4 is available to tolerate all the faults, so the reliability cost is zero. By this definition, for a specific mapping pattern when l links have errors, the reliability cost is given by

$$R_i^l = \sum_{\mathrm{SD}} R_i^{\mathrm{SD}} P_i^{\mathrm{SD}} \tag{7}$$

where R_i^{SD} is the reliability cost of each path from source node S to destination node D, and P_i^{SD} is the binary function to tell if there is a communication between S and D, as defined in (4). Similar to the energy model, R_i^l is a discrete variable on the number of faulty links l and condition i. By considering all the faulty scenarios, the accuracy of the model is ensured. The reliability cost of an NoC for a specific mapping pattern is then expressed as

$$R = \sum_{l=0}^{L} \sum_{i=1}^{M} R_i^l P_i \tag{8}$$

where P_i is defined as that in (5) and

$$RN = R/RN' \tag{9}$$

where RN' is the reliability cost when the number of faulty links reaches the maximum value that can be tolerated; it is used in (9) as a normalization factor.

By substituting (6) and (9) into (1), the total cost for a mapping pattern can be estimated. This cost model is applicable to various different NoC topologies and routing algorithms. The reasons are as follows. First, previous models require to know either the communication path in advance [21], which is only limited to deterministic routing algorithm, or the Manhattan distance of the source–destination pair [6], [20], which may not work in topologies other than mesh. In contrast, in this paper, only the number of routers and links on the communication path is required in the energy estimation. The energy is measured dynamically, so the evaluation can be done as soon as the communication path changes. In this way, the energy can be evaluated no matter which topology and routing algorithm the NoC is. Second, the reliability is enhanced by mapping the source–destination pair to make the bounding box close to a square [21]. However, this requirement may not be satisfied in topologies other than a mesh. The proposed method only needs to ensure that the communication path can avoid faulty links, which boarded the use of NoC topology. Finally, the proposed approach considers the joint optimization of both energy and reliability. Efforts are made in [6] and [20] to repair faulty PEs with minimum energy consumption, but no quantitative analysis has been presented. In our model, both the energy and reliability cost are considered as discrete variables and the cost is evaluated with their probabilities. This ensures that the measurement is independent of the NoC topology and routing algorithm, thus achieving an accurate measurement of overall energy and reliability. To conclude, the PCM is not only flexible, but also energy and reliability aware.

B. Performance Model

The performance measures, such as throughput and latency, are also important metrics of a system and cannot be neglected when energy and reliability are optimized in the mapping process. Consequently, in addition to the unified cost model, a performance model is also proposed. As latency is greatly influenced by the congestion in the network, to avoid congestion in each node is an effective way to reduce latency. At the same time, less congestion can lead to higher throughput. Therefore, the performance constraint is considered to be given by the bandwidth requirement, which is closely related to the congestion [7]. The bandwidth constraints are given in (10) and they are exploited to balance the communication volumes in every node, such that the congestion can mostly be avoided to guarantee the performance including both throughput and latency

$$\sum_{a_{ij}} \left[f(P_{\mathrm{map}(c_i),\mathrm{map}(c_j)}, l_{ij}) \times V^{ij} \right] \le B(l_{ij}) \tag{10}$$

where $B(l_{ij})$ is the bandwidth of the link l_{ij}. The binary function f indicates whether the link l_{ij} is utilized by the path $P_{\mathrm{map}(c_i),\mathrm{map}(c_j)}$, as defined by

$$f(P_{\mathrm{map}(c_i),\mathrm{map}(c_j)}, l_{ij}) = \begin{cases} 0, & l_{ij} \notin P_{\mathrm{map}(c_i),\mathrm{map}(c_j)} \\ 1, & l_{ij} \in P_{\mathrm{map}(c_i),\mathrm{map}(c_j)}. \end{cases} \tag{11}$$

IV. Energy and Reliability Oriented Mapping

A. Problem Definition

Using the models described previously, the problem of an energy- and reliability-aware mapping under certain performance constraints is defined as follows.

Given an APCG and an NoC of routers and PEs with any topology and routing algorithm, finding a mapping function

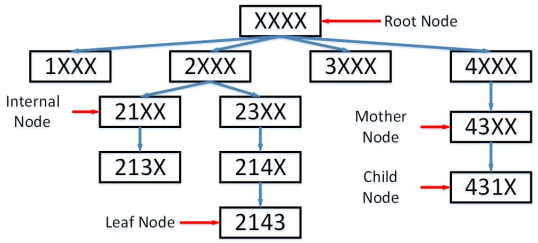


Fig. 3. Structure of a search tree.

map() that maps an IP core $c_i \in C$ in the APCG to a PE in the NoC such that the following conditions are satisfied:

Min: Cost $= (1-\alpha)EN + \alpha RN$

s.t. $\text{map}(c_i) \neq \text{map}(c_j) \quad \forall c_i \neq c_j \in C$

$\sum_{a_{ij}} [f(P_{\text{map}(c_i),\text{map}(c_j)}, l_{ij}) \times V^{ij}] \leq B(l_{ij})$.

B. BBPCR Mapping

1) Branch- and Bound-Based Mapping: A branch and bound (BB)-based algorithm efficiently maps a given application onto PEs interconnected by an NoC [7]. The best mapping with the minimum energy consumption is found by establishing a search tree, as shown in Fig. 3. The root node represents the mapping that none of the IPs in the APCG is mapped onto the PEs. The internal nodes indicate the partial map with some IPs mapped onto some of the PEs and a leaf node represents the complete mapping that all IPs are mapped. In the search tree, a child node is based on its mother node. When building the searching tree, the BB algorithm uses the upper bound cost (UBC) and lower bound cost (LBC) of each internal node to decide whether this node should be created or not. For each node, if its LBC is larger than its minimal UBC, the node will be discarded and all the child nodes based on this node will be discarded too; otherwise, this node will be created for further comparison. In this way, a large number of candidate mappings that are not likely to be the best mapping pattern can be discarded early without calculating its cost; thus, the computational overhead can be reduced.

In the BB algorithm, the UBC of an internal node is defined as the cost of a leaf node created by a temporary greedy mapping of the remaining IP cores. For an internal node, the LBC is divided into three parts, as shown in the following:

$$\text{LBC} = \text{LBC}_{m,m} + \text{LBC}_{u,u} + \text{LBC}_{m,u} \tag{12}$$

where $\text{LBC}_{m,m}$ is the partial cost calculated by (1) for the mapped IP cores. $\text{LBC}_{u,u}$ is the partial cost due to communications among the cores that are not mapped yet. It is calculated by assuming the closest possible locations of the unmapped cores. $\text{LBC}_{m,u}$ is the partial cost caused by the communication among the already mapped and unmapped cores. It is also calculated by assuming the best mapping of the unmapped cores onto the unoccupied PEs with respect to the mapped ones.

2) BBPCR: As shown in Fig. 3, an internal node closer to the root node incurs a larger computational load. Therefore, if those internal nodes with a shorter distance to the root node are discarded as many as possible, the best mapping can be found more efficiently. However, the BB algorithm treats all the internal nodes equally and this will incur a large computational complexity. In this paper, a mapping algorithm using BBPCR is used to expedite the process of finding the best mapping pattern by differentiating the priorities of the internal nodes. Special care is also taken to protect those optimal candidates from undesired deletion during the acceleration of the searching. As defined in (13), the ratio between two adjacent partial costs is utilized to improve the efficiency of discarding the candidate nodes that cannot lead to the best mapping

$$\text{ratio}_{l+1,l} = \frac{\text{Cost}^{l+1}}{\text{Cost}^{l}}. \tag{13}$$

Taking (5), (6), (8) and (9) into (13) gives a specific expression, as shown in the following:

$$\text{ratio}_{l+1,l} = \frac{\sum_{i=1}^{\binom{L}{l+1}} \left((1-\alpha)E_i^{l+1}/EN' + \alpha R_i^{l+1}/RN'\right)P_i}{\sum_{i=1}^{\binom{L}{l}} \left((1-\alpha)E_i^{l}/EN' + \alpha R_i^{l}/RN'\right)P_i}. \tag{14}$$

In (14), the faulty probabilities of the links are independent of each other. The proposed model is applicable to links with multiple faulty probabilities. To demonstrate the nonunified faulty probabilities, an example of two values, p and p_a, is made in this paper for simplicity. The difference of faulty probabilities is based on the communication volumes passing through the link because large volumes of data passing lead to high energy consumption. Hotspots in thermal mainly results from high energy consumption, and this high temperature is more likely to cause the errors in links [28], [29]. Therefore, when the link is in the high communication volumes region, p_j is considered to be p_a; otherwise, it is considered to be p. This is a simple example to show the variance of faulty probabilities and the proposed model can be integrated with more sophisticated thermal model.

Then P_i in (14) is substituted by p_a and p. Referring to (3), E_i^l is determined by the path of each source–destination pair for a given mapping pattern. Therefore, E_i^l will not change much as l changes. When l changes, the variation of R_i^l is only limited to a small set of paths, so $\text{ratio}_{l+1,l}$ can be estimated by (the explicit derivation is included in the Appendix)

$$\text{ratio}_{l+1,l} < \frac{L-l}{l+1}\frac{1}{(1-p)^2}. \tag{15}$$

Since p is typically less than 0.5, $\text{ratio}_{l+1,l}$ decreases rapidly as l increases. When l is large enough, $\text{ratio}_{l+1,l}$ becomes very small. The lth partial cost can be expressed by

$$\text{Cost}^l = \text{Cost}^0 \times \prod_{k=1}^{l} \text{ratio}_{k,k-1}. \tag{16}$$

Here Cost^0 is the cost of the mapping pattern when no links are faulty. In this way, Cost^l is small enough to be ignored when l is large enough. Accordingly, the condition for deleting as many as possible the internal nodes that are close to the root node is set by

$$\text{LBC} > \min\{\text{UBC}\} \times (1 + \text{ratio}_{1,0}). \tag{17}$$

TABLE I
COMPUTATION COMPLEXITY COMPARISONS

Optimistic case	
BB	$o(2^{n+1} \times n^3)$
BBPCR	$o(n^5)$
Typical case	
BB	$o(n^3) \times (\frac{k^{n+1}-k}{(k-1)^2} - \frac{n}{k-1})$
RCRBB	$o(n^3) \times (\frac{(k-1)^{n+1}-k+1}{(k-2)^2} - \frac{n}{k-2})$
Pessimistic case	
BB	$o(n^3) \times \sum_{j=0}^{n-1} \prod_{i=0}^{j}(n-i)$
BBPCR	$o(n^3)\{\sum_{j=1}^{n-1}[\prod_{i=1}^{j-1}(n-1)](n-j)^2+n\}$

For each internal node, once this condition is met, this node and all its child nodes are discarded; otherwise, the internal node is saved as a candidate node for further comparison.

3) Simplification of Partial Cost Ratio Evaluation: As discussed in the Section II, computational overhead needs to be constrained when optimizing energy and reliability during mapping. In this paper, special care is taken to reduce the computational complexity of BBPCR.

The computational complexity of BBPCR is mainly due to two sources: the first is the comparison between all candidate mapping patterns, which has been simplified in the previous subsection and the second is the computation of $\text{ratio}_{1,0}$ for each internal node. In this subsection, a simplification method is utilized to reduce the complexity in calculating $\text{ratio}_{1,0}$ for each internal node. $\text{ratio}_{1,0}$ is the ratio between the cost when one link is faulty and the cost when no link is faulty; therefore, it does not change significantly for different mapping patterns. Consequently, it is estimated as the average value of the ratios for a set of different mapping patterns instead of including all the possible mapping patterns. In this way, the computational complexity for this ratio is significantly reduced.

To show the reduction of computational complexity, it is compared with the BB mapping algorithm. Since it is difficult to find out the accurate number of the internal nodes and when these internal nodes can be discarded, three different conditions are considered for estimating the complexity: an optimistic case, a typical case and a pessimistic case. In the optimistic case with the least computation load, it is assumed that the majority of the internal nodes can be discarded and only one node at each branch of the search tree is left. The other extreme is the pessimistic case as it is expected that in the worst scenario, only one node can be discarded for each branch. This results in the most complex computation. In the more likely typical case, it is assumed that k nodes are left at each branch. Compared with the BB algorithm, the proposed algorithm focuses on discarding more nodes that are closer to the root node, so in all three cases, the BB algorithm is expected to have at least one more node left in each branch than BBPCR. In this paper, the number of the loops is considered as a measure for the computational complexity, because in each loop, the computation is quite simple and the complexity is similar. Hence, the computational complexity for three cases is compared in Table I, where n is the number of PEs in a reconfigurable architecture.

For reconfigurable architectures of different sizes, Fig. 4 shows the reduction in computational complexity of the proposed approach in comparison with BB. As can be seen, this reduction is positive in all cases, which indicates that the proposed method reduces the computational complexity of a mapping. It can also be seen that the reduction increases when the size of the NoC becomes larger. This indicates that the proposed method is preferable for use in a large NoC system. Further experiments confirm this theoretical analysis (as shown in the next section).

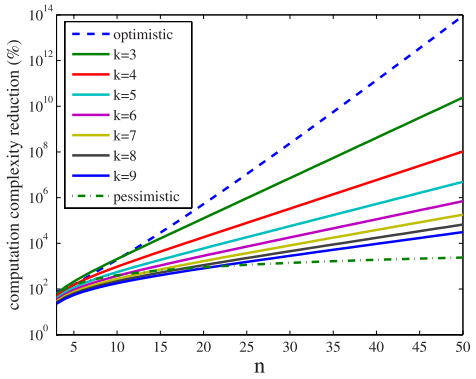


Fig. 4. Computational complexity reduction of BBPCR with respect to BB.

TABLE II
EXPERIMENTAL ENVIRONMENT OF THE ROUTER

Switch technology	Wormhole
Arbitration policy	Roll-turn
Virtual channel	Use virtual channels
E_{Rbit}	4.171 nJ/bit [31]
E_{Lbit}	0.449 nJ/bit [31]

V. EXPERIMENTAL RESULTS

As discussed in Section III-A, the proposed approach has a high flexibility for various NoC topologies and routing algorithms. In this section, four different combinations of NoC topology and routing algorithm are implemented to show the flexibility of the proposed approach. In addition, this approach is also compared with the state-of-the-art approaches [6], [20], [21] and a classical algorithm in terms of energy, reliability, performance and computational complexity.

The best mapping pattern is found using a C++ program and the time consumed can be obtained as well. Using the best mapping pattern, simulations are performed on a cycle accurate simulator implemented by SoC Designer [30]. In the simulator, each node of the NoC is constructed by a router and a PE. The experimental environment of a router is shown in Table II. Our mapping method is independent of the switch technology, the arbitration policy, and the use of virtual channels. The experimental environment of the router is chosen to be the same as those in [21] for a fair comparison. $E_{R\text{bit}}$ and $E_{L\text{bit}}$ are the energy consumed by 1 bit of data transporting through a router and a link, as given in an open source simulator [31].

For all of the following simulations, the experiment conditions are identical. The energy is computed by counting the number of links and routers in the communication path. The total energy obtained from that is then divided by

TABLE III
TOPOLOGY AND ROUTING ALGORITHM COMBINATIONS USED FOR THE DEMONSTRATION OF FLEXIBILITY

NoC	Topology		Routing Algorithm	
	Name	Category	Name	Category
1	Torus[34]	Mesh/Torus	Odd-even	Adaptive
2	Spidergon[35]	Ring	CrossFirst	Deterministic
3	de Bruijn Graph[36]	Custom	Deflection	Deterministic
4	Mesh	Mesh/Torus	Full-adaptive	Adaptive

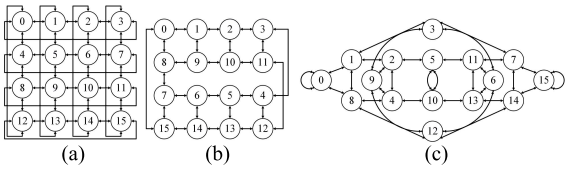


Fig. 5. Examples of three different topologies with 16 nodes. (a) Torus. (b) Spidergon. (c) Binary de Bruijn Graph.

TABLE IV
BENCHMARKS USED IN THE SIMULATION

Benchmark	Number of IPs	Application
DVOPD[38]	32	Dual video object plane decoder
VOPD[39]	16	Video object plane decoder
MPEG4[40]	9	MPEG4 decoder
PIP[39]	8	Picture in picture
MWD[39]	12	Multi-window display
mp3enc mp3dec[41]	13	Mp3 encoder & mp3 decoder
263enc mp3dec[41]	12	H.263 encoder & mp3 decoder
263dec mp3dec[41]	14	H.263 decoder & mp3 decoder
H.264 [42]	14	H.264 decoder
HEVC [43]	16	High Efficiency Video Coding decoder
Freqmine [44]	12	Data mining application
Swaption [44]	15	Computes portfolio prices using Monte-Carlo simulation
random1[45]	16	Generated by TGFF
random2[45]	16	Generated by TGFF

the total flits to estimate the average energy of a mapping pattern [7]. Although this method does not return the accurate value of energy consumption, it provides an effective means to compare the energy consumption by different mapping patterns. Link errors are randomly injected with the probability p and p_a and the reliability of a mapping pattern is then calculated by accurately evaluating the probability of a flit transportation [32].

A. Flexibility

To demonstrate the flexibility of the proposed approach, different combinations of topology and routing algorithm can be utilized for experiments. A survey of 60 papers that includes 66 topologies and 67 routing algorithms [33] shows that 56.1% of the networks are mesh/torus, 12.1% are custom, and 7.6% are ring. It also shows that 62.7% of the networks use a deterministic routing and the remaining 37.3% use an adaptive routing. Therefore, four common topologies and routing algorithms are selected for the simulation as shown in Table III. For the adaptive routing algorithm, both partial and fully adaptive routing algorithms are considered. Odd–even is chosen as an example of partial adaptive, while force-directed wormhole routing [37] is chosen for fully adaptive one. The chosen topologies are also illustrated with a 16-node example in Fig. 5.

Fourteen commonly used application benchmarks, as shown in Table IV, are mapped onto the NoC described above. Twelve out of the fourteen benchmarks are from real applications. The other two are random benchmarks generated by task graphs for free [45]. To show the flexibility, it is not required to set a weight on energy or reliability, so the weight parameter α is set to 0.5. The failure probabilities of links are assumed to have two values considering different communication volumes passing the links, as discussed in Section IV. When the probability of a link being faulty is larger than 0.5, the NoC is unlikely to operate normally, so the probability of fault is chosen to be no larger than 0.5 (i.e., $p\_a = 0.5, 0.5, 0.1, 0.01$, and 0.001 and $p = 0.5, 0.1, 0.01, 0.001$, and 0.0001). Due to the lack of literature on mappings applicable to different topologies and routing algorithms, the best mapping found by BBPCR is compared with a classical algorithm simulated annealing (SA) [46]. SA is a probabilistic method to find the global minimum of a cost function, and it can be considered as a reference method which indicates the quasi-global optimal solution. SA is just a mapping approach and it is flexible but not energy and reliability aware. Therefore, the experiments compare the best mappings found by BBPCR and SA both using PCM proposed Section III.

For the four combinations of NoC topologies and routing algorithms in Table III, all the 14 benchmarks in Table IV are successfully mapped onto the NoCs with BBPCR. The computation time consumed by BBPCR and SA to find the best mapping is considered as an indicator for the computational complexity. For each best mapping, the communication energy consumption and the reliability also are simulated with respect to different probabilities of faulty links. These comparison results of BBPCR and SA are all shown in Figs. 6–10. It shows that the BBPCR outperforms the SA in terms of energy, reliability, and computation time averagely. Statistic results show that 75% of the optimum mapping found by SA is worse than BBPCR by up to 97% in terms of reliability. Although the energy consumption of less than a half of optimum mappings generated by BBPCR is slight larger than SA by no more than 10%, the time of searching for the best mapping consumed by SA is more than 80 times of that of BBPCR in average. The reason is that SA searches for the best mapping with minimum cost calculated by (1) at the expense of searching time. In this way, less than a half of the best mappings found by BBPCR are of larger cost than that of SA by 3.5% averagely. For these cases, the energy consumption is slightly larger than SA, but some of them are still of better reliability compared with SA. Meanwhile, the rest of best mappings found by BBPCR have considerable advantages in terms of cost compared with SA. For these cases, both energy consumption and reliability are superior to SA. To summarize, the average energy reduction, reliability enhancement, and the ratio of computation time of the optimum mappings found by BBPCR are better than or at least the same as that obtained by SA for different probabilities of faulty links, as shown in Fig. 11. These results show that the amount of energy reduction changes little with the decrease of the probability of

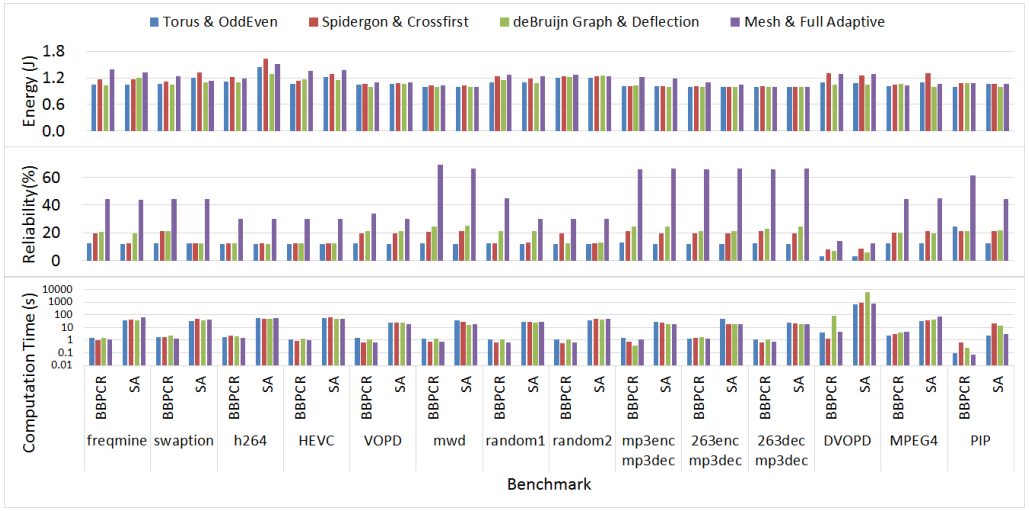


Fig. 6. Comparison of energy consumption, reliability, and computation time between BBPCR and SA when $p = 0.5$ and $p\_a = 0.5$.

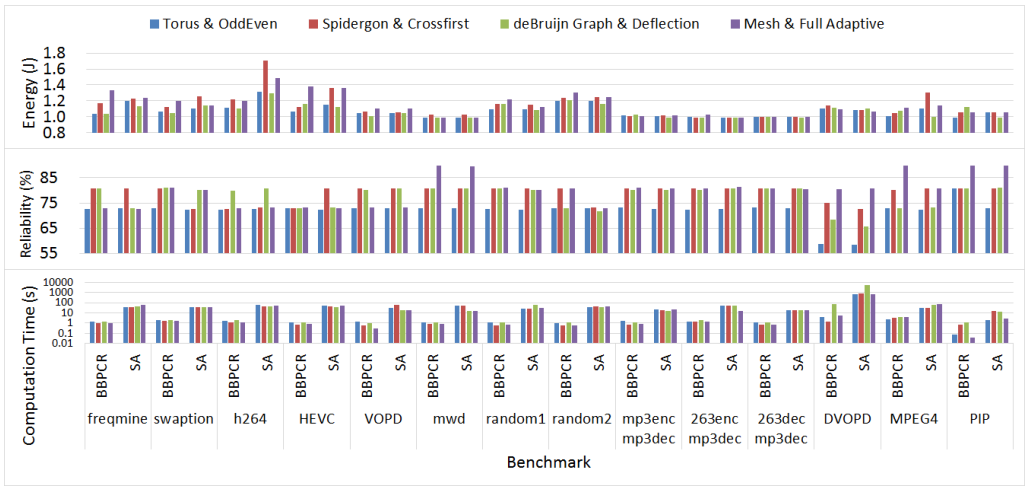


Fig. 7. Comparison of energy consumption, reliability, and computation time between BBPCR and SA when $p = 0.1$ and $p\_a = 0.5$.

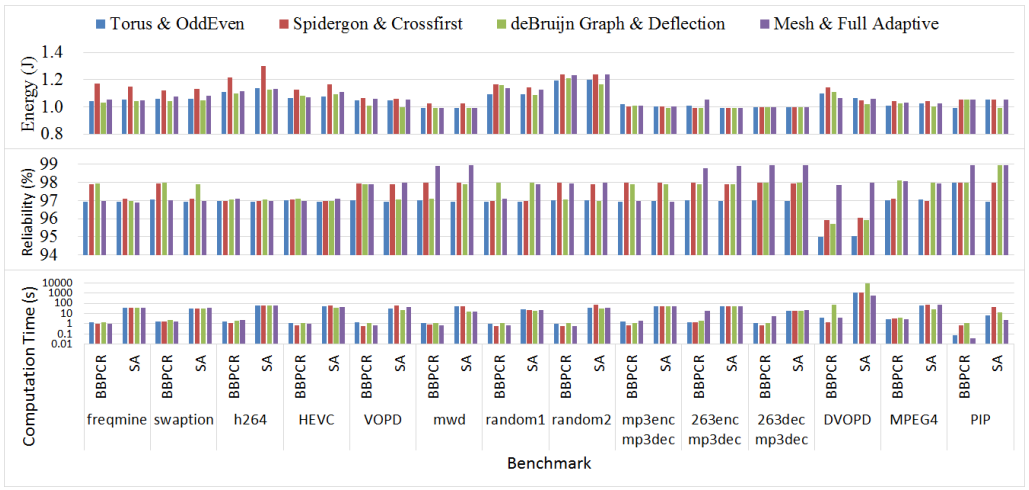


Fig. 8. Comparison of energy consumption, reliability, and computation time between BBPCR and SA when $p = 0.01$ and $p\_a = 0.1$.

a faulty link. As shown in Fig. 11(b), the reliability improvement decreases with the probability of a faulty link because all mapping patterns are highly reliable when p is small. In terms of computation time, BBPCR spends less time dealing with the faulty links when the probability of a faulty link decreases, and therefore, the time of finding the best mapping

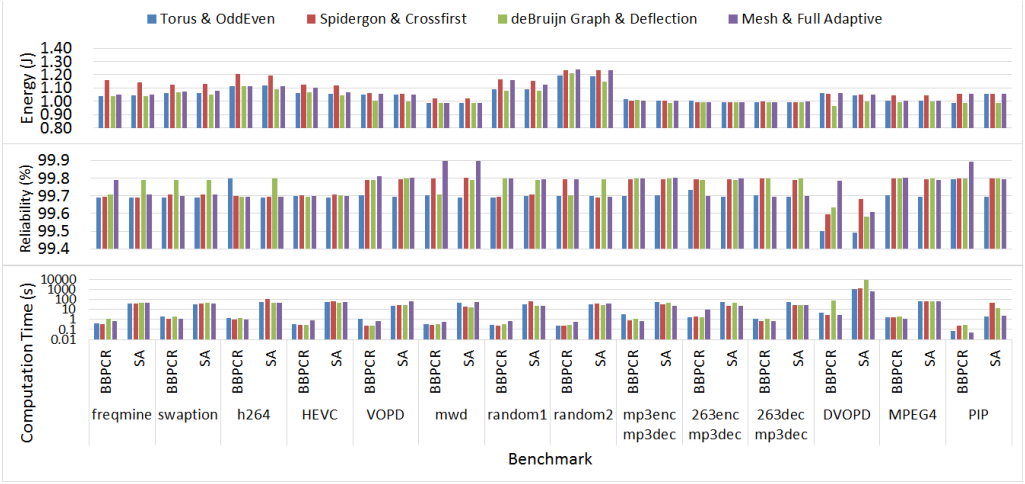


Fig. 9. Comparison of energy consumption, reliability, and computation time between BBPCR and SA when $p = 0.001$ and $p\_a = 0.01$.

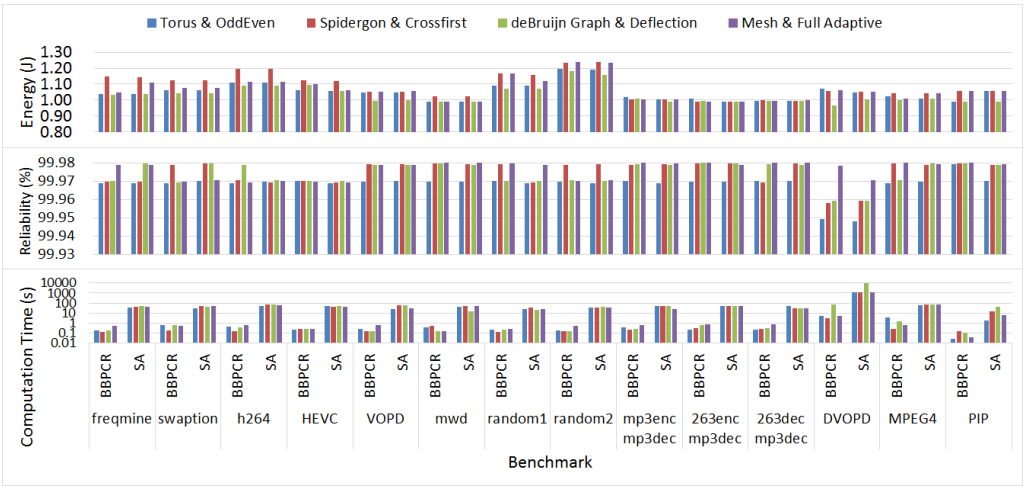
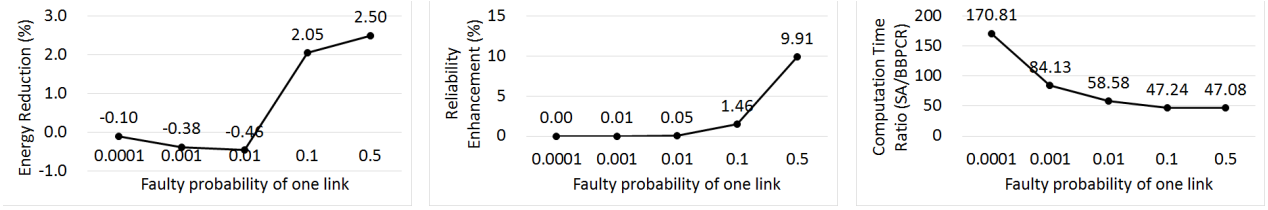


Fig. 10. Comparison of energy consumption, reliability, and computation time between BBPCR and SA when $p = 0.0001$ and $p\_a = 0.001$.



(a) (b) (c)

Fig. 11. Average energy reduction, reliability enhancement, and ratio of computation time with respect to the probability of a faulty link(p). (a) Average energy reduction. (b) Average reliability enhancement. (c) Average ratio of computation time.

decreases accordingly. Unlike BBPCR, the computation time of SA is independent of the probability of a faulty link, so in most cases the computation time remains the same. Hence, the difference in computation time between BBPCR and SA increases when p becomes smaller, as shown in Fig. 11(c).

All these results show that the proposed BBPCR approach is applicable to various NoC topologies and routing algorithms. The proposed cost model, PCM, is independent of and, therefore, applicable to any type of NoC topologies and routing algorithms. This ensures the flexibility of the proposed mapping approach.

B. Comparison to the State of the Art

As our approach has been confirmed to be highly flexible for various topologies and routing algorithms, the BBPCR approach is compared with other state-of-the-art methods in this section. This evaluation includes energy, reliability,

TABLE V
AVERAGE COMMUNICATION ENERGY, PERFORMANCE, AND EXECUTION TIME OF THE PROPOSED BBPCR, [6], AND [20] IN COMPARISON AGAINST OPTIMAL MAPPING

	BBPCR	[6]	[20]
Execution time speedup vs. optimal solution (%)	99.34	99.12	93.46
Communication energy overhead vs. optimal solution (%)	4.12	23.70	9.10
Performance overhead vs. optimal solution (%)	7.08	24.37	13.18

TABLE VI
CHARACTERISTICS OF BENCHMARKS

Benchmark	Number of IPs	Min/Max communication
mpeg4[40]	9	1/942
telecom[7]	16	11/71
ami25[47]	25	1/4
ami49[47]	49	1/14
H.264	14	3280/124417508
HEVC	16	697/1087166
Freqmine	14	12/6174
Swaption	15	145/747726417

performance, and computational complexity. As discussed previously, the methods in [6], [20], and [21] have considered both energy and reliability, but they are only applicable to a 2-D mesh topology with deterministic routing algorithms. In other words, they are all energy and reliability aware, but not flexible. The experimental results in [20] give the overhead comparison with global optimal mappings for both [6] and [20]. The applications used in these experiments include MEPG4 decoder, video object plane decoder, 263 decoder, 263 encoder and Mp3 encoder. They all consist of less than 9 IPs and therefore can be mapped on 3×3 NoC of mesh topology. In this case, the computation time for the exhaustive searching is bearable. Therefore, the best mappings found by BBPCR for the same applications are also compared with the global optimal mappings. The average speedup, energy consumption overhead, and reliability overhead for these three methods are listed in Table V. It can be observed clearly that the proposed BBPCR has the largest speed up but the smallest energy and reliability overhead among these three methods. In other words, these mappings found by BBPCR are the ones that are the closest to the global optimal mappings.

For the BB algorithm in [21], a series of in-depth comparisons with more complicated benchmarks are made with the benefit of open source code. Since all the benchmarks used here have more than nine IPs, the computation time for exhaustive searching becomes highly demanding. Instead, SA is utilized to indicate the margin between the quasi-global optimal solution and results obtained by BB and BBPCR. For a fair comparison, the same routing algorithm and topology are used in the simulations, as those in [21]. Besides the test cases in [21], four additional benchmarks of high communication load volume and computation complexity are also included for further comparison, as shown in Table VI. The same assumption about link failure possibility is made as the experiment of flexibility in Section V-A. Extensive experiments are carried out to evaluate the best mapping obtained by BBPCR, BB, and SA with respect to energy consumption, reliability, throughput, and latency. In addition, the time consumed by finding the optimum mapping is also compared. The runtime is also a good indicator to the computational complexity of each method, which is important to consider for the dynamic reconfiguration during run time. Table VII summarizes the overall results of these experiments including the average, minimum, and maximum values of two weight parameters selected from the results in [21]. As shown in Table VII, on average, BBPCR shows significant advantages in all aspects compared with BB. Statistic results show that the energy consumption of less than 45% of the best mappings obtained by BBPCR is not as good as those of SA. In terms of reliability, 70% of best mappings found by BBPCR are better than SA by up to 43%. In average, both the latency and throughput of the best mappings found by BBPCR outperform those of SA. More importantly, the simulation time consumed by SA is far more than BBPCR and ratio between them ranges between 17.2 and 1667.9. Table VIII compares the results generated by BB and SA and it shows that in general BB is further away from quasi-global optimum solutions SA compared with BB. To conclude, BBPCR makes greater tradeoff between energy, reliability, throughput, latency, and run time than the other two algorithms (BB and SA). The detailed experiments for $\alpha = 0.2$ is discussed in the following.

1) Energy Reduction: Fig. 12 compares the energy consumption of the best mapping patterns with respect to different probabilities of faulty links and the x-axis is the p, as defined in the Section V-A. For all eight benchmarks, the energy consumed by the mapping patterns found by BBPCR is much less than that found by BB. The best mapping found by BBPCR consumes almost the same energy as SA does, but on average, the energy still reduces by 5.82%.

2) Reliability Enhancement: The reliability of the best mapping patterns found by BBPCR, BB, and SA are compared in Fig. 13. In general, the reliability of BBPCR is better than the other two. For the benchmarks ami25 and ami49, the reliabilities are quite similar for BB and BBPCR. The reason is that the communication volumes of ami25 and ami49 are relatively smaller than the other six benchmarks (in Table VI), and therefore, less communication path is used. In this case, when the errors are generated randomly, an alternative path can be easily found. A reliability enhancement of such NoC mappings is quite difficult. Even for the other six benchmarks, the reliability enhancement is also quite small when $p \leq 0.04$. When the probability of a faulty link is very small, the reliability of any mapping pattern is very high, so little difference can be obtained between the proposed approach and BB. Meanwhile, the reliability improvement becomes rather obvious when $p \geq 0.0625$, which confirms the advantage of the proposed approach. To summarize, approximately an average of 8.69% enhancement in reliability is obtained using BBPCR. The reliability of the best mapping pattern found by BBPCR is quite similar to that of SA, which means that results obtained by BBPCR are quite close to the quasi-global optimal solutions.

3) Performance: As discussed in Section III-B, performance constraints are imposed to ensure the performance of the

TABLE VII
SUMMARY OF COMPARISONS OF BBPCR AGAINST BB AND SA

	$\alpha = 0.2$						$\alpha = 0.6$					
	Against BB			Against SA			Against BB			Against SA		
	Max	Min	Avg.	Max	Min	Avg.	Max	Min	Avg.	Max	Min	Avg.
Energy Reduction	46.25%	1.8%	23.56%	32.6%	-8.91%	5.82%	55.48%	0%	19.72%	32.6%	-17.4%	3.25%
Reliability Enhancement	113.31%	0%	8.69%	43.65%	-4.43%	2.86%	32.14%	-0.06%	3.91%	43.65%	4.08%	2.12%
Throughput Improvement	21.43%	-2.78%	8.91%	18.18%	-5.41%	5.77%	16.13%	0%	8.37%	18.18%	-2.78%	5.77%
Latency Reduction	48.11%	-12%	11.98%	23.99%	-13.18%	5.68%	42.21%	-11.3%	10.02%	22.38%	-12.8%	4.28%
Computation Time Ratio (divided by BBPCR)	11.8×	1×	3.47×	1167.9×	17.2×	326.4×	11.8×	1×	2.75×	1656×	17.7×	292.93×

TABLE VIII
SUMMARY OF COMPARISONS OF BB AGAINST SA

	$\alpha = 0.2$			$\alpha = 0.6$		
	Max	Min	Avg.	Max	Min	Avg.
Energy Reduction(%)	22.00	86.98	-29.11	21.59	-157.1	-31.16
Reliability Enhancement(%)	17.81	-44.76	-3.89	19.09	-24.29	-1.54
Throughput Improvement(%)	5.88	-12.5	-2.71	4.55	-11.43	-2.38
Latency Reduction(%)	23.87	-79.12	-11.89	11.04	-71.04	-9.73
Computation Time Ratio (divided by BB)(×)	496	9.84	111.23	496	17.6	120.6

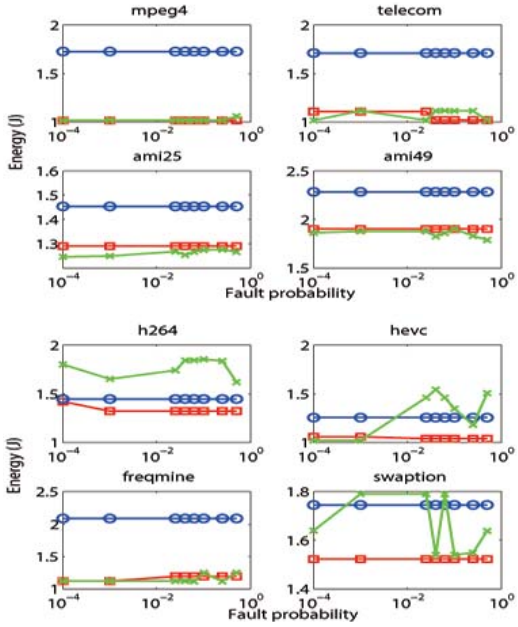


Fig. 12. Comparison of energy consumption of the best mapping found by BBPCR (red), BB (blue), and SA (green) for different benchmarks with respect to p.

obtained mapping when optimizing energy and reliability. The performance evaluation includes two measures: throughput and latency. Fig. 14 shows the throughput comparison among BBPCR, BB, and SA. These comparisons clearly show that BBPCR is superior to the other mapping algorithms in terms of the throughput in most cases. As one of the most important metrics for a mapping pattern, an average increase in throughput by 8.91% and 5.77% is achieved compared with BB and SA, respectively.

Since the latency of a specific mapping is closely related to the throughput, the latency of a mapping pattern obtained by BBPCR, BB, and SA for each benchmark is compared with respect to throughput using $p = 0.01$ as an example. As shown

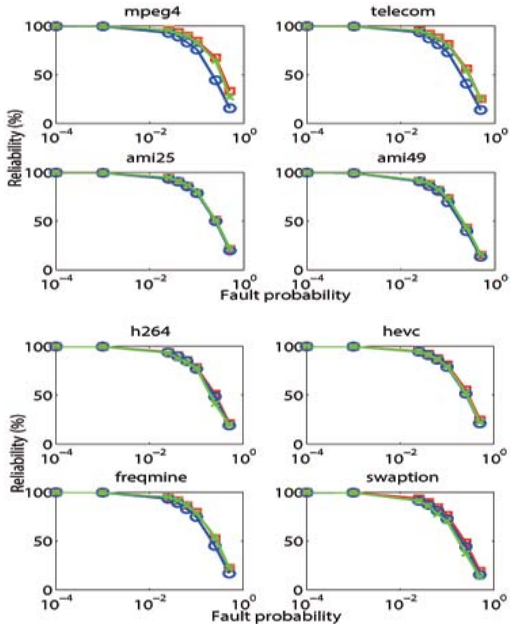


Fig. 13. Comparison of reliability of the best mapping found by BBPCR (red), BB (blue), and SA (green) for different benchmarks with respect to p.

in Fig. 15, when the throughput is small, the latency for each benchmark remains 20 cycles/flit approximately. However, as the throughput increases, the network comes into saturation, thus resulting in a latency wall. The average latency of the BBPCR outperforms that of BB by 11.98%.

4) Run-Time Reduction: The run time of finding the best mapping pattern is used to approximate the computational complexity of each mapping approach. The program is run on a platform, as described in Table IX.

Fig. 16 shows the run time to find best mapping patterns for BBPCR, BB, and SA. For all the benchmarks but mpeg4, BBPCR takes much less time to find the desired mapping compared with BB. As for mpeg4, since there are only 9 IPs, the run times for both BB and BBPCR are very short and similar. With an increased number of IPs, for example, with 25 IPs in ami25 and 49 IPs in ami49, the proposed approach shows a significant advantage in the run time of finding the best mapping pattern. This result is consistent with the theoretical analysis in previous sections. In summary, more than 200% speed up has been obtained using BBPCR on average. Although SA can find the better mapping pattern than BBPCR in some certain cases as discussed above, the time consumed by SA is averagely 300× of BBPCR uses, and to

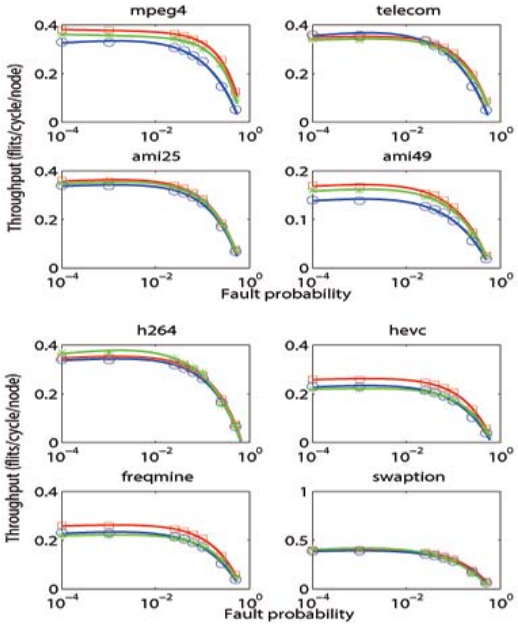


Fig. 14. Comparison of throughput of the best mapping found by BBPCR (red), BB (blue), and SA (green) for different benchmarks with respect to p (markers: simulated data and line: fitting curve).

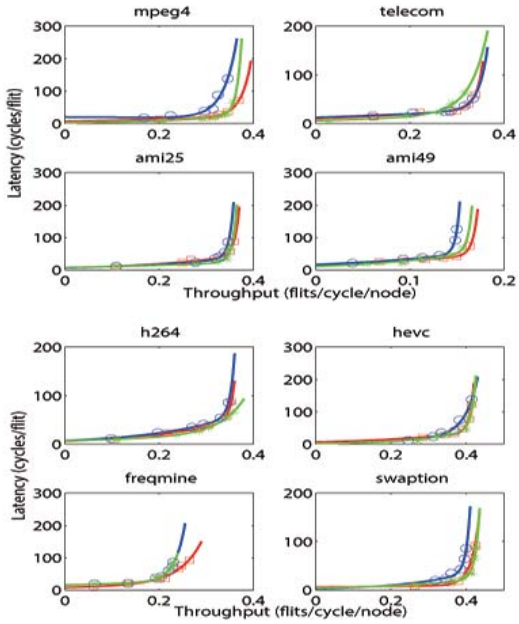


Fig. 15. Comparison of latency of the best mapping found by BBPCR (red), BB (blue), and SA (green) when $p = 0.01$ for different benchmarks (markers: simulated data and line: fitting curve).

TABLE IX
PLATFORM FOR RUNNING SIMULATIONS

CPU	2 × Inter(R) Xeon(R) E5520
Main Frequency	2.27GHz
Memory	16GB
Operating System	Linux

be more explicit, it takes SA about 2 h to find an optimum mapping in some cases. This is unacceptable in a dynamically reconfigurable system.

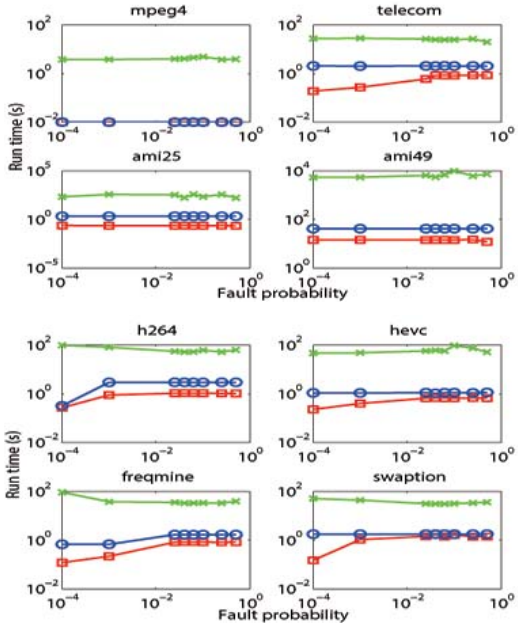


Fig. 16. Comparison of run time for finding the best mapping for different benchmarks (red: BBPCR, blue: BB, and green: SA).

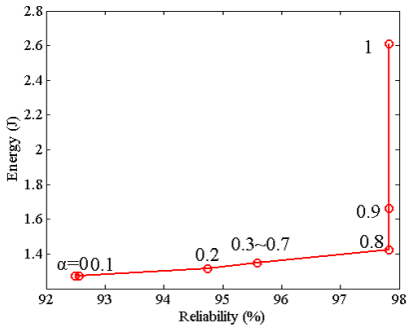


Fig. 17. Energy and reliability change with α changes for benchmark ami25 when $p = 0.025$ and $p\_a = 0.01$.

C. Discussion of α and Area

1) Discussion of α: The influence of weight parameter α is discussed to present the tradeoff between energy consumption and reliability. An example of benchmark ami25 when $p = 0.025$, $p\_a = 0.01$ is shown in Fig. 17. As expected, experimental results show that when α increases, the reliability is monotonic increasing. In the meanwhile, energy consumption is also increasing as the sacrifice of reliability enhancement. Therefore, a tradeoff between energy and reliability has to be made in the real application and this can be achieved by choosing a proper value of α.

2) Discussion of Area: The area of NoC used in this paper is estimated and provided for fabrication cost reference. All the mapping approaches (BBPCR, BB, and SA) have the same NoC and router structure, and therefore, the fabrication cost is the same as well. A typical NoC router always has the following six components: routing computation logic, virtual channel allocation logic, switch allocation logic, crossbar, input/output buffer, and interrouter link. As shown in Table X, the area of fabricating such a router will approximately be 0.33 mm^2 with the technology of 65-nm Taiwan Semiconductor Manufacturing Company. Moreover, compared with the

TABLE X
SCALE AND AREA ESTIMATION OF A ROUTER

component	Scale (10,000 gates)	Area (mm^2)
I/O Channel	4.6	0.073
Crossbar	0.9	0.014
Virtual Channel Controller	14.5	0.232
Arbiter	0.7	0.011
Total	20.7	0.33

Intel's 80-core NoC [48], whose router area is approximately 0.34 mm^2, the fabrication cost in terms of area in this paper is not large. The total tie area of the biggest NoC used in this paper of 49 cores is about 169 mm^2, as die area of the Intel's 80-core NoC is 275 mm^2, whose core contains a router and a processor element.

VI. CONCLUSION

When dealing with the problem of finding the best application mapping pattern, it is of great importance to select the appropriate metrics for evaluation. In this paper, a highly flexible and accurate cost model PCM with energy and reliability awareness is proposed. Based on this model, an efficient approach BBPCR is created for finding the optimum mapping solution with minimal energy consumption and maximal reliability. To the best of our knowledge, the proposed method is the only one with both flexibility as well as energy and reliability awareness. In addition to that, experimental results show that for a certain mapping found by BBPCR still outperforms other state-of-the-art approaches in terms of energy, reliability, performance, and efficiency.

The proposed mapping approach is independent of the detailed structure of an NoC system, so it is applicable to a broad category of topologies and routing algorithms. This is a unique feature of the proposed approach and it is particularly useful in the application mapping of reconfigurable architectures. The performance in terms of throughput and latency is guaranteed by posing bandwidth constraints. However, this could be further improved by measuring both latency and throughput with quantitative models. In this way, a comprehensive cost model could then be developed for optimizing various metrics of energy consumption, reliability, and performance (for both latency and throughput). Moreover, the faulty probability of each link is assumed to be two values as an example in this paper. In addition, the faulty probability for each link is differentiated by considering different communication volumes for simplicity. A more accurate model of faulty links, such as considering the hotspots, can also be included in the future work.

APPENDIX

$$\mathrm{Cost}^l = \sum_{i=1}^{\binom{L}{l}} \left[\frac{(1-\alpha)E_i^l}{EN'} + \frac{\alpha R_i^l}{RN'} \right] P_i$$

$$\mathrm{Cost}^l = \xi^l \sum_{i=1}^{\binom{L}{l}} P_i = \xi^l \sum_{i=0}^{n} \binom{n}{i}\binom{L-n}{l-i} p^l (1-p)^{L-l} \left[\frac{p\_a(1-p\_a)}{p(1-p)} \right]^i$$

where n means there are n links whose fault probability is p_a, and ξ^l means the average cost. Then

$$\begin{aligned}
\mathrm{ratio}_{l+1,l} &= \frac{\mathrm{Cost}^{l+1}}{\mathrm{Cost}^l} \\
&= \frac{\xi^l + 1}{\xi^l} \frac{\sum_{i=0}^{n} \binom{n}{i}\binom{L-n}{l+1-i} p^{l+1}(1-p)^{L-l-1} \left[\frac{p_a(1-p_a)}{p(1-p)} \right]^i}{\sum_{i=0}^{n} \binom{n}{i}\binom{L-n}{l-i} p^l (1-p)^{L-l} \left[\frac{p_a(1-p_a)}{p(1-p)} \right]^i} \\
&= \frac{\xi^{l+1}}{\xi^l} \times \frac{p}{1-p} \times \frac{\sum_{i=0}^{n} \binom{n}{i}\binom{L-n}{l+1-i} \left[\frac{p_a(1-p_a)}{p(1-p)} \right]^i}{\sum_{i=0}^{n} \binom{n}{i}\binom{L-n}{l-i} \left[\frac{p_a(1-p_a)}{p(1-p)} \right]^i}.
\end{aligned}$$

As $p < p\_a < 0.5$, so

$$1 < \frac{p\_a(1-p\_a)}{p(1-p)} < \frac{1}{4p(1-p)}$$

and then

$$\begin{aligned}
\mathrm{ratio}_{l+1,l} &< \frac{\xi^{l+1}}{\xi^l} \times \frac{p}{1-p} \times \frac{\binom{L}{l+1} \frac{1}{4p(1-p)}}{\binom{L}{l}} \\
&= \frac{\xi^{l+1}}{\xi^l} \times \frac{p}{1-p} \times \frac{L-l}{l+1} \times \frac{1}{4p(1-p)}.
\end{aligned}$$

Then

$$\mathrm{ratio}_{l+1,l} < \frac{\xi^{l+1}}{\xi^l} \times \frac{L-l}{l+1} \times \frac{1}{4(1-p)^2} \approx \frac{L-l}{l+1} \times \frac{1}{4(1-p)^2}.$$

Therefore, we can get (15) from (13).

REFERENCES

[1] N. W. Bergmann, S. K. Shukla, and J. Becher, "QUKU: A dual-layer reconfigurable architecture," *ACM Trans. Embedded Comput. Syst.*, vol. 12, no. 1s, Mar. 2013, Art. ID 63.

[2] J. A. Walker, M. A. Trefzer, S. J. Bale, and A. M. Tyrrell, "PAnDA: A reconfigurable architecture that adapts to physical substrate variations," *IEEE Trans. Comput.*, vol. 62, no. 8, pp. 1584–1596, Aug. 2013.

[3] J. S. Kim, M. B. Taylor, J. Miller, and D. Wentzlaff, "Energy characterization of a tiled architecture processor with on-chip networks," in *Proc. Int. Symp. Low Power Electron. Design*, 2003, pp. 424–427.

[4] A. B. Kahng, B. Li, L.-S. Peh, and K. Samadi, "ORION 2.0: A fast and accurate NoC power and area model for early-stage design space exploration," in *Proc. Design, Autom., Test Eur. Conf. Exhibit.*, Apr. 2009, pp. 423–428.

[5] W. J. Dally and B. P. Towles, *Principles and Practices of Interconnection Networks*. San Francisco, CA, USA: Morgan Kaufmann, 2004.

[6] C.-L. Chou and R. Marculescu, "FARM: Fault-aware resource management in NoC-based multiprocessor platforms," in *Proc. Design, Autom., Test Eur. Conf. Exhibit.*, Mar. 2011, pp. 1–6.

[7] J. Hu and R. Marculescu, "Energy- and performance-aware mapping for regular NoC architectures," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 24, no. 4, pp. 551–562, Apr. 2005.

[8] Y.-C. Chang, C.-T. Chiu, S.-Y. Lin, and C.-K. Liu, "On the design and analysis of fault tolerant NoC architecture using spare routers," in *Proc. 16th Asia South Pacific Design Autom. Conf.*, Jan. 2011, pp. 431–436.

[9] A. Kohler, G. Schley, and M. Radetzki, "Fault tolerant network on chip switching with graceful performance degradation," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 29, no. 6, pp. 883–896, Jun. 2010.

[10] M. Suess, R. Trautner, R. Vitulli, J. Ilstad, and D. Thurnes, "Technical Dossier on on-board payload data processing," ESA, Paris, France, Tech. Rep. TECEDP/2011.110/Ms, 2011.

[11] R. Trautner, "ESA's roadmap for next generation payload data processors," in *Proc. Data Syst. Aerosp. Conf. (DASIA)*, 2011, pp. 159–161.

[12] M. Shafto *et al.*, "Modeling, simulation, information technology & processing roadmap," NASA, Washington, DC, USA, Tech. Rep. 11, Apr. 2012.

[13] L. Ost *et al.*, "Power-aware dynamic mapping heuristics for NoC-based MPSoCs using a unified model-based approach," *ACM Trans. Embedded Comput. Syst.*, vol. 12, no. 3, Mar. 2013, Art. ID 75.

[14] M. Mandelli *et al.*, "Energy-aware dynamic task mapping for NoC-based MPSoCs," in *Proc. IEEE Int. Symp. Circuits Syst.*, May 2011, pp. 1676–1679.

[15] C. Çelik and C. F. Bazlamaçci, "Energy and buffer aware application mapping for networks-on-chip with self similar traffic," *J. Syst. Archit.*, vol. 59, no. 10, pp. 1364–1374, 2013.

[16] S. Tosun, "New heuristic algorithms for energy aware application mapping and routing on mesh-based NoCs," *J. Syst. Archit.*, vol. 57, no. 1, pp. 69–78, Jan. 2011.

[17] A. Das, A. Kumar, and B. Veeravalli, "Reliability-driven task mapping for lifetime extension of networks-on-chip based multiprocessor systems," in *Proc. Design, Autom., Test Eur. Conf. Exhibit.*, Mar. 2013, pp. 689–694.

[18] A. Das and A. Kumar, "Fault-aware task re-mapping for throughput constrained multimedia applications on NoC-based MPSoCs," in *Proc. 23rd IEEE Int. Symp. Rapid Syst. Prototyping*, Oct. 2012, pp. 149–155.

[19] X. Qi, D. Zhu, and H. Aydin, "Global reliability-aware power management for multiprocessor real-time systems," in *Proc. IEEE 16th Int. Conf. Embedded Real-Time Comput. Syst. Appl.*, Aug. 2010, pp. 183–192.

[20] F. Khalili and H. R. Zarandi, "A reliability-aware multi-application mapping technique in networks-on-chip," in *Proc. 21st Euromicro Int. Conf. Parallel, Distrib., Netw.-Based Process.*, Feb./Mar. 2013, pp. 478–485.

[21] C. Ababer, H. S. Kia, O. P. Yadav, and J. Hu, "Energy and reliability oriented mapping for regular networks-on-chip," in *Proc. 5th IEEE/ACM Int. Symp. Netw. Chip*, May 2011, pp. 121–128.

[22] R. Marculescu, U. Y. Ogras, L.-S. Peh, N. E. Jerger, and Y. Hoskote, "Outstanding research problems in NoC design: System, microarchitecture, and circuit perspectives," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 28, no. 1, pp. 3–21, Jan. 2009.

[23] M. Valinataj, S. Mohammadi, J. Plosila, P. Liljeberg, and H. Tenhunen, "A reconfigurable and adaptive routing method for fault-tolerant mesh-based networks-on-chip," *AEU-Int. J. Electron. Commun.*, vol. 65, no. 7, pp. 630–640, Jul. 2011.

[24] J. Jiao and Y. Fu, "Exploiting and evaluating the potentials of the link addition method for NoC transient error mitigation," *Microprocessors Microsyst.*, vol. 38, no. 3, pp. 183–196, May 2014.

[25] H.-W. Lee, K. Lee, and E. Modiano, "Maximizing reliability in WDM networks through lightpath routing," in *Proc. IEEE Global Telecommun. Conf.*, Dec. 2011, pp. 1–6.

[26] K. Lee, H.-W. Lee, and E. Modiano, "Reliability in layered networks with random link failures," *IEEE/ACM Trans. Netw.*, vol. 19, no. 6, pp. 1835–1848, Dec. 2011.

[27] T. T. Ye, L. Benini, and G. De Micheli, "Analysis of power consumption on switch fabrics in network routers," in *Proc. 39th Design Autom. Conf.*, 2002, pp. 524–529.

[28] R. Blish *et al.*, "Critical reliability challenges for the international technology roadmap for semiconductors (ITRS)," Int. Sematech, Austin, TX, USA, Tech. Rep. 03024377A-TR, 2003.

[29] S. Murali, *Designing Reliable and Efficient Networks on Chips*. New York, NY, USA: Springer-Verlag, 2009.

[30] (2012). *SoC Designer*. [Online]. Available: http://www.carbondesignsystems.com/soc-designer-plus/

[31] *Reliable-NoC Tool*. (2010). [Online]. Available: http://venus.ece.ndsu.nodak.edu/ cris/software.html

[32] A. Patooghy, H. Tabkhi, and S. G. Miremadi, "RMAP: A reliability-aware application mapping for network-on-chips," in *Proc. 3rd Int. Conf. Dependability*, Jul. 2010, pp. 112–117.

[33] E. Salminen, A. Kulmala, and T. D. Hamalainen, "Survey of network-on-chip proposals," in *Proc. OCP-IP*, Mar. 2008, pp. 1–13.

[34] G.-M. Chiu, "The odd-even turn model for adaptive routing," *IEEE Trans. Parallel Distrib. Syst.*, vol. 11, no. 7, pp. 729–738, Jul. 2000.

[35] E. Wachter, A. Erichsen, A. Amory, and F. Moraes, "Topology-agnostic fault-tolerant NoC routing method," in *Proc. Design, Autom., Test Eur. Conf. Exhibit.*, Mar. 2013, pp. 1595–1600.

[36] H. Moussa, A. Baghdadi, and M. Jezequel, "Binary de Bruijn on-chip network for a flexible multiprocessor LDPC decoder," in *Proc. 45th ACM/IEEE Design Autom. Conf.*, Jun. 2008, pp. 429–434.

[37] T. Schonwald, J. Zimmermann, O. Bringmann, and W. Rosenstiel, "Fully adaptive fault-tolerant routing algorithm for network-on-chip architectures," in *Proc. 10th Euromicro Conf. Digit. Syst. Design Archit., Methods, Tools (DSD)*, Aug. 2007, pp. 527–534.

[38] K. Samadi, "Accurate estimators and optimizers for networks-on-chip," Ph.D. dissertation, Dept. Elect. Eng., Univ. California, San Diego, CA, USA, 2010.

[39] D. Bertozzi *et al.*, "NoC synthesis flow for customized domain specific multiprocessor systems-on-chip," *IEEE Trans. Parallel Distrib. Syst.*, vol. 16, no. 2, pp. 113–129, Feb. 2005.

[40] E. B. van der Tol and E. G. T. Jaspers, "Mapping of MPEG-4 decoding on a flexible architecture platform," *Proc. SPIE, Media Processors*, vol. 4674, pp. 1–13, Dec. 2002.

[41] K. Srinivasan, K. S. Chatha, and G. Konjevod, "Linear programming based techniques for synthesis of network-on-chip architectures," in *Proc. IEEE Int. Conf. Comput. Design, VLSI Comput. Processors*, Oct. 2004, pp. 422–429.

[42] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 560–576, Jul. 2003.

[43] G. J. Sullivan, J. Ohm, W.-J. Han, and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1649–1668, Dec. 2012.

[44] (2009). *PARSEC*. [Online]. Available: http://parsec.cs.princeton.edu/overview.htm

[45] R. P. Dick, D. L. Rhodes, and W. Wolf, "TGFF: Task graphs for free," in *Proc. 6th Int. Symp. Hardw./Softw. Codesign*, Mar. 1998, pp. 97–101.

[46] S. Chai, Y. Li, J. Wang, and C. Wu, "A list simulated annealing algorithm for task scheduling on network-on-chip," *J. Comput.*, vol. 9, no. 1, pp. 176–182, Jan. 2014.

[47] (2007). *MCNC Benchmarks*. [Online]. Available: http://vlsicad.eecs.umich.edu/BK/MCNCbench

[48] Y. Hoskote, S. Vangal, A. Singh, N. Borkar, and S. Borkar, "A 5-GHz mesh interconnect for a teraflops processor," *IEEE Micro*, vol. 27, no. 5, pp. 51–61, Sep./Oct. 2007.

Leibo Liu received the B.S. degree in electronic engineering from Tsinghua University, Beijing, China, in 1999, and the Ph.D. degree from the Institute of Microelectronics, Tsinghua University, in 2004.

He is currently an Associate Professor with the Institute of Microelectronics, Tsinghua University. His current research interests include reconfigurable computing, mobile computing, and VLSI DSP.

Chen Wu received the B.S. degree from the School of Micro-Electronics and Solid-State Electronics, University of Electronic Science and Technology of China, Chengdu, China, in 2012. He is currently pursuing the master's degree with the Institute of Microelectronics, Tsinghua University, Beijing, China.

His current research interests include fault-tolerant system design, reliability modeling, and reconfigurable networks-on-chip.

Chenchen Deng received the B.S. degree in electronic engineering from the Beijing University of Posts and Telecommunications, Beijing, China, in 2007, and the Ph.D. degree in engineering science from the University of Oxford, Oxford, U.K., in 2012.

She is currently a Post-Doctoral Research Fellow with the Institute of Microelectronics, Tsinghua University, Beijing. Her current research interests include reconfigurable computing, 3-D IC, and system-on-a-chip design.

Shouyi Yin received the B.S., M.S., and Ph.D. degrees in electronic engineering from Tsinghua University, Beijing, China, in 2000, 2002, and 2005, respectively.

He was with Imperial College London, London, U.K., as a Research Associate. He is currently with the Institute of Microelectronics, Tsinghua University, as an Associate Professor. He has authored over 20 refereed papers. His current research interests include mobile computing, wireless communications, and system-on-a-chip design.

Dr. Yin served as a TPC Member or reviewer for international key conferences and leading journals.

Qinghua Wu was born in Sichuan, China, in 1990. He received the B.E. degree from the Institute of Automation Engineering, Sichuan University, Chengdu, China, in 2008, and the M.E. degree from the Institute of Microelectronics, Tsinghua University, Beijing, China, in 2013.

His current research interests include reliability modeling and evaluation, fault-tolerant system design, and reconfigurable networks-on-chip.

Jie Han received the B.Sc. degree in electronic engineering from Tsinghua University, Beijing, China, in 1999, and the Ph.D. degree from the Delft University of Technology, Delft, The Netherlands, in 2004.

He is currently an Assistant Professor with the Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB, Canada. His current research interests include reliability, fault tolerance and energy efficiency, nanoelectronic circuits and systems, and novel computational models for nanoscale and biological applications.

Shaojun Wei was born in Beijing, China, in 1958. He received the Ph.D. degree from the Faculte Polytechnique de Mons, Mons, Belgium, in 1991.

He became a Professor at the Institute of Microelectronics, Tsinghua University, Beijing, in 1995. His current research interests include VLSI system-on-a-chip design, EDA methodology, and communication application-specified integrated circuit design.

Prof. Wei is a Senior Member of the Chinese Institute of Electronics.