IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received July 26, 2021, accepted August 1, 2021, date of publication August 3, 2021, date of current version August 12, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3102270

Rapid Topology Generation and Core Mapping of Optical Network-on-Chip for Heterogeneous Computing Platform

YONG WOOK KIM[1], (Student Member, IEEE), SEO HONG CHOI[2], (Student Member, IEEE), AND TAE HEE HAN[3], (Member, IEEE)

[1]Department of Electrical and Computer Engineering, Sungkyunkwan University, Suwon 16419, South Korea
[2]Department of Artificial Intelligence, Sungkyunkwan University, Suwon 16419, South Korea
[3]Department of Semiconductor Systems Engineering, Sungkyunkwan University, Suwon 16419, South Korea

Corresponding author: Tae Hee Han (than@skku.edu)

This work was supported by Samsung Electronics Company Ltd. under Grant IO201209-07877-01.

ABSTRACT The explosive growth of deep learning (DL)–based artificial intelligence (AI) applications necessitates extraordinary computing capabilities that cannot be achieved using traditional CPU standalone computing. Therefore, the heavy mission-critical DL kernel computing currently relies on a heterogeneous computing (HGC) platform integrated with CPUs, GPUs, and accelerators, as well as substantial data storage elements. However, the metallic electrical interconnection in the existing manycore platform would not be sustainable for handling the massively increasing bandwidth demand of big data driven AI applications. Incorporating an optical network-on-chip (ONoC) for providing ultrahigh bandwidth, we propose a rapid topology generation and core mapping of ONoC (REGO) for energy-efficient HGC multicore architecture. The genetic algorithm (GA)-based REGO utilizes the structural characteristics of the optical router to the fitness function and thus compromises the trade-off between the required throughput, optical signal-to-noise ratio (OSNR), and total energy consumption. Furthermore, the crossover step accelerates the convergence speed by suppressing randomness in the GA, thus significantly reducing excessive running time owing to the NP-hard property. The generated ONoC through REGO demonstrates, on an average, an increase of 63.29 % and 22.80 % in throughput and a decrease of 50.24 % and 9.56 % in energy per bit, in the VGG-16 and VGG-19 compared with the conventional mesh- and torus-topology-based ONoCs, respectively.

INDEX TERMS Deep learning kernel, genetic algorithm, heterogeneous computing platform, topology generation, optical network-on-chip.

I. INTRODUCTION

Deep learning (DL), a class of machine learning algorithms, trains a nonlinear function approximator represented by a deep neural network (DNN) architecture using input-output pairs of training data [1]. The primary goal of DL is to improve accuracy by learning the weights through backward propagation of errors (backpropagation). Repetitive operations that occur while learning errors in backpropagation require extremely high parallelism and vector-matrix operations. Therefore, a heterogeneous computing (HGC) platform that combines various types of processors and dedicated accelerators is required instead of a legacy CPU-based architecture [2]. In addition, an ultra-wideband on-chip network infrastructure is essential for handling excessively heavy data traffic.

Network-on-chip (NoC) is a scalable solution for on-chip communication infrastructure that can handle the ever-increasing processor cores integrated on a single chip. However, despite the continuing progress in transistor miniaturization, the challenging problems in the backend-of-the-line (BEOL) fabrication steps that form the interconnect layer using metallic interconnects impede the expansion of the on-chip communication bandwidth. An optical NoC (ONoC) based on silicon photonics is being actively investigated as an alternative to electrical NoCs (ENoCs). Semiconductor industries such as IBM, Intel, and Mellanox have developed several optical devices and interface technologies which can be deployed in ONoCs [3]–[5]. In addition, AyarLabs has developed TeraPHY, a highly

The associate editor coordinating the review of this manuscript and approving it for publication was Massimo Cafaro.

integrated smart photoelectric chiplet capable of operating with a bandwidth of several tens of Tb/s in an ASIC, CPU, and FPGA package in a recent study [6]. In [7], the authors compared the electrical mesh (EMesh) with four different types of ONoCs, including their own LumiNOC, under fair simulation conditions. All NoCs under comparison were organized in 64 tiles comprising corresponding core and router pairs operating at 5GHz using a 22nm CMOS process technology library. Table 1 shows a comparison of ENoC with four different ONoC architectures in an experimental environment established by Cheng. Corona, Flexishare, and Clos have 26, 3, and 1.5 times higher throughput and 14, 5, and 24 times higher power efficiency compared with EMesh, respectively. LumiNOC, which is an application-specific ONoC architecture, achieved 34 times higher throughput per watt compared to EMesh. In summary, ONoC is still far from mass production; nonetheless, it has the potential to provide a significantly higher bandwidth and energy efficiency compared to that of the ENoC. Therefore, a rapid topology generation and core mapping of an ONoC (REGO) is proposed to effectively assess the unique DNN traffic patterns and HGC architecture.

TABLE 1. Comparison of ENoC and ONoCs with regard to power consumption and throughput.

	Classification	Throughput (Tbps)	Throughput per power (Tbps/W)
Emesh [8]	ENOC	3.0	0.1
Corona [9]	ONoC	73.6	1.4
FlexiShare [10]	ONoC	9.0	0.5
Clos [11]	ONoC	4.1	2.4
LumiNOC [7]	ONoC	8.0	3.4

Primarily, irregular topology generation and core mapping are NP-hard problems that require a tremendous computational load. Furthermore, to minimize the laser power consumption of an ONoC, a large number of worst-case optical signal-to-noise ratio (OSNR) calculations with dynamically varying optical signal propagation models in various routing candidate paths must be involved. The resonance structure of micro-ring resonators (MRs) requires large-scale iterative calculations until the optical signal is stabilized in each optical router. The computational complexity increases exponentially with system augmentations along with the memory footprint [12]. Therefore, a high-speed algorithm that can achieve irregular topology generation and core mapping of ONoCs that satisfies a given design goal within a reasonable time is essential.

For the design space exploration of NoC, particle swarm optimization (PSO) [13] and genetic algorithm (GA) [14], a type of meta-heuristic, are commonly used. However, the PSO has three drawbacks when applied to topology generation in NoCs. Because PSO finds the optimal solution only through the direction toward the current best solution, 1) it is affected more by the initial initialization population than GA, and 2) has the drawback of increasing the chance of falling into the local minima problem [15]. 3) The network constraints for topology generation result in a huge computational load on the process of changing the location of particles in the PSO.

For these reasons, PSO is mainly used for core and application mapping [16]–[19]. The GA is a parallel and global optimization problem solving technique that mimics natural selection and genetic inheritance [20]. The GA is frequently used in environments where the best solution must be found within an acceptable time [21]. Several studies have been conducted for GA-based irregular topology generation and core mapping to optimize power and performance under highly variable data traffic environments [22]–[24].

The GA is a population-based optimization heuristic that finds a solution through the iteration of selection, crossover, and mutation steps from the initial population of a group of chromosomes. The GA heuristic has been widely adopted for NP-hard problems in architecture space exploration including network topology and core mapping techniques. Three drawbacks are usually mentioned in the discussion of GA. First, GA is prone to a local minima problem unless it randomly generates an initialization population for genetic diversity. Second, an in-depth consideration of the fitness function in the evolution phase is required for the accuracy and convergence speed of the algorithm. Finally, it is difficult to assign an optimization problem to genetic data, such as chromosomes and genes. To alleviate the first and second drawbacks, the links of routers are randomly generated to maximize the randomness of the initial population in the initialization phase of the REGO, and the fitness function is defined to reflect the ONoC design characteristics. Meanwhile, because the throughput and OSNR are significantly affected by the internal connectivity and core mapping in the ONoCs, the objectives of the fitness function and the basic elements of the GA can be properly adapted to the searching scheme for ONoC topology solutions. The third drawback can thus be resolved. Therefore, the GA was selected as a framework to determine the optimal topology and core mapping solution for the ONoC implementation of the HGC platform.

While an electrical router consists of symmetric crossbar switches, an optical router comprising waveguides and MR switches is configured asymmetrically to minimize insertion loss and crosstalk noise. It is widely known that the OSNR dominates the power consumption of ONoCs and strongly depends on insertion loss and crosstalk noise. Furthermore, the insertion loss and crosstalk noise are significantly affected by the number of ports of the optical router according to the arrangement of optical elements [25]. Therefore, REGO considers OSNR variations based on the optical element configuration in the fitness calculation to optimize the trade-off between the data throughput and power consumption in HGC platform.

Furthermore, this study focuses on accelerating the convergence speed by applying the crossover that reflects the structural change of the routers depending on the OSNR and the number of ports. In the crossover step of the REGO, a fitness variation-based genetic data exchange scheme is applied to avoid unnecessary searching for chromosomes which violate ONoC constraints. Core mapping refers to the method of allocating cores to a given NoC topology in a specific order. The topology generation arranges the links of the routers and allocates appropriate routers according to the number of ports in the arranged network. In this process, the aggregation of router connections implies the location of cores, thus allowing topology generation to incorporate core mapping of ONoC with a small computational burden. In addition, considering different types of processor cores that show different capabilities and characteristics in an irregular topology, core mapping must be jointly performed with topology generation to meet the design objectives. Consequently, the consolidated topology generation and core mapping of the REGO are natural and beneficial for jointly optimizing the design objective with given constraints.

The remainder of this paper is organized as follows. Related work and background are described in Section II. Section III presents the main algorithm flow of GA-based REGO and describes how the properties of the optical elements are applied to the technique. The simulation results and analysis under various conditions are described in Section IV. Finally, the conclusions are drawn in Section 5.

II. RELATED WORK

A. GA-BASED ENoC TOPOLOGY GENERATION

Core mapping has been studied for optimizing the power and performance of the target application in a regular topology where the topology is predetermined for routing efficiency and scalability. A variety of core mapping schemes for optimizing power consumption and performance in NoCs have been conducted [26]–[28]. In [26], a core mapping technique of NoC using reinforcement learning in an HGC platform without a test set prepared in advance was proposed. In addition, Tahir *et al.* proposed a congestion-aware core mapping scheme using betweenness centrality that can identify highly loaded NoC links in [27]. However, these studies were derived from a lightweight computing algorithm that simply swaps the location of the core, and it is difficult to apply it to a topology generation method that requires consideration of various network conditions.

Existing studies on GA-based ENoC topology generation have mainly focused on minimizing power consumption or maximizing throughput. Leary *et al.* proposed a GA-based topology generation for application-specific ENoC [29]. Herein, the authors achieved 30% lower total power consumption than deterministic heuristic techniques by considering the system-level floorplan with wire-length constraints along with the power consumption due to physical links. In [30], the power consumption and router resources were minimized while meeting bandwidth constraints through GA-based floorplan-aware topology synthesis. In addition, a GA-based mapping and routing (GAMR) approach was proposed for low energy design of ENoCs under bandwidth constraints [22]. The GAMR automatically mapped the cores of a given application onto the ENoC and generated deterministic deadlock-free minimal routing paths.

In contrast, GA-based topology generation schemes have been suggested for application specific ENoCs that have been pursued to improve throughput [23], [31]. These studies considered the required throughput of the given applications in the fitness function and evolution phase of the GA. Although the ENoC topology generation techniques mentioned so far can be partially adapted to ONoCs, additional considerations are essential because of the fundamentally different signal characteristics and interconnection medium. Moreover, the worst-case OSNR calculation is mandatory to determine the minimum required laser source power, which dominates the overall power consumption and performance.

B. ONOC ARCHITECTURE FOR HGC PLATFORM

Studies on irregular-topology-based ENoCs have been extended to accommodate optical interconnection thus achieving ultra-high bandwidth required for handling ever-increasing big data and/or DNN acceleration. Ahmed *et al.* proposed PHENIC 3D-ONoC, a silicon photonic 3D-NoC architecture for heterogeneous many-core system-on-chips (MCSoCs) [32]. The PHENIC 3D-ONoC is composed of an electronic control network (ECN) for path reservation, which can configure optical routers, and a number of photonic communication networks (PCNs), thereby providing approximately 10 % improvement in throughput compared to conventional 2D-mesh based ENoC. In [33], SHARP (shared heterogeneous architecture with reconfigurable photonic network-on-chip) showed 34 % more throughput and 25 % less energy consumption per bit compared to the mesh-based ENoC. SHARP clusters CPU and GPU cores around the same router and dynamically allocated bandwidth between CPU and GPU cores through single-writer multiple-reader (SWMR) crossbars according to the application requirement. Although PHENIC 3D-ONoC and SHARP deployed full optical data paths for exploiting the low power and broadband advantages of the optical interconnect, the potential capabilities of irregular topologies were not addressed.

With the explosive growth of big data-based DL applications, diverse approaches in terms of architectural aspects to satisfy the enormous throughput and energy efficiency are actively progressing. For optical signal detection, the ONoC laser source sets the power margin considering the OSNR, photodetector sensitivity, and laser wall-plug efficiency (L_e). MR heater for resonance wavelength tuning is reported to account for approximately 20 % of the total power consumption of the various ONoC topologies [34]. While the sensitivity and L_e of the photodetector are hardware constraints that are not controllable, the OSNR and the number of MR heaters that significantly affect the total power

consumption of ONoCs have strong correlation with the implementation style. Consequently, both OSNR and the number of MR heaters must be assessed in the process of GA-based topology generation and core mapping.

III. REGO

Fig. 1 depicts the overall procedure of GA-based REGO. First, three types of inputs are defined: ONoC parameters, GA parameters, application task graph. The ONoC parameters include the system-level ONoC specification such as configurable router types and loss coefficient of optical elements. The GA parameters indicate the constants required for initialization and evolution such as population size, selection probability, and the fractional ratio of crossover. The REGO receives as inputs an application task graph including the number of cores and ONoC parameters, which further includes the available router structure and loss and noise factors of the optical elements. Thus, the REGO can accommodate various router structures and optical elements because it calculates the worst-case OSNR through loss and noise parameters obtained in advance through the parameters of optical routers and elements.

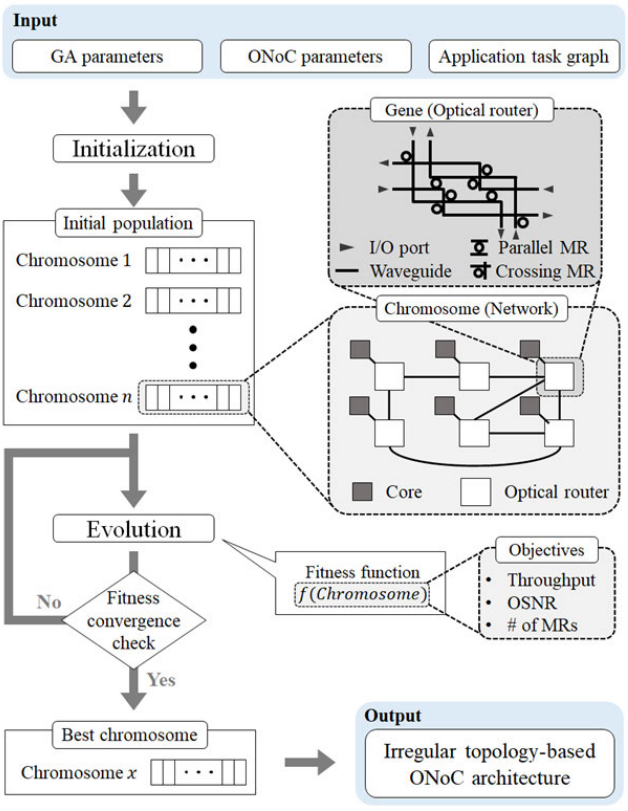

FIGURE 1. Overall procedure of REGO.

A gene, which is a primitive element of the GA, is mapped with a router, including structure and link information connected to the adjacent routers and the corresponding core. Next, the genes are gathered to form a chromosome corresponding to the entire ONoC topology. All chromosomes go through the initialization phase, which creates a random population. Then enter the evolution phase with genetic information containing the fractional ratio accounting for the number of chromosomes classified into selection, crossover, and mutation. For each iteration of the evolution phase, the fitness of all chromosomes is calculated by the fitness function. The objectives of the fitness function include features of an ONoC such as OSNR and the total number of MRs. When the convergence condition is satisfied, the best population is obtained from the REGO which implies an irregular topology-based ONoC solution.

While the initialization phase attempts to maximize randomness, the fitness function and evolution phase consider the characteristics of the optical elements to optimize the performance and energy efficiency. The REGO finds a fitness-based solution by incorporating the crucial information relating to the OSNR, throughput, and MR heaters into the fitness function thereby optimizing both throughput and power consumption. In addition, unreliable factors caused by improper router connections can be reduced by attempting to crossover with regard to fitness variations of the objectives.

A. PROBLEM DEFINITION AND TERMINOLOGY

The GA-based REGO for the HGC platform satisfies the following two constraints to ensure path validity between the connected routers:

- Constraint 1. All routers and cores in a chromosome must be guaranteed to be connected.
- Constraint 2. A direct network in which each core forms a pair with only a single router is mandatory.

Table 2 describes the notations used in the REGO. The gene, an element of the set $\mathbb{G}$, represents a single router with connectivity information. Each chromosome represents the

TABLE 2. Notations of REGO.

Notation	Description
TG	Task graph = {$\mathbb{D}$ (data traffic), $\mathbb{V}$(vertices)}
$\mathbb{V}$	$\{v_i \mid v_i$ is i^{th} core$\}$
$\mathbb{R}$	$\{(r_i, r_i) \mid r_i$ is i^{th} router connected with $v_i\}$
NRP	$\{nrp_i \mid nrp_i$ is number of router ports of $r_i\}$
SP	Signal Path = $\{sp_{ij} : v_i \rightarrow v_j$ with data traffic $d_{ij} \mid v_i(source), v_j(destination) \in \mathbb{V}$ and $d_{ij} \in \mathbb{V}\}$
RT	Routing Table = $\{rt_{ij} \mid rt_{ij}$ is set of routers in $np_{ij} \in \mathbb{R}\}$
$\mathbb{G}$	Set of genes = $\{(gl_i, gt_i) \mid gl_i$ is set of router links of r_i and gt_i is type of $r_i\}$
$\mathbb{C}$	Chromosome = {$\mathbb{G}$, RT}
$\mathbb{C}_{set}$	Chromosome set = $\{\mathbb{C}_i \mid \mathbb{C}_i$ is i^{th} chromosome of gene $\mathbb{G}_i\}$
λ	Fractional ratio of chromosomes for selection in $\mathbb{C}_{set}$
ξ	Fractional ratio of chromosomes for crossover in $\mathbb{C}_{set}$
μ	Fractional ratio of chromosomes for mutation in $\mathbb{C}_{set}$
p_s	Selection probability in selection step
p_{min}	Minimum number of ports of configurable router types
p_{max}	Maximum number of ports of configurable router types

entire ONoC topology as a group of genes and a routing table. Fig. 2 depicts an example configuration of a single chromosome that allows routers with four and five ports. Chromosomes have connectivity and routing table information for all routers in the network. Connectivity indicates the router number connected to each router r_i and the corresponding port number. The routing table stores the input ports, output ports, and MR control signals for the source node v_i and destination node v_j.The cardinality of $\mathbb{R}$ is the same as the number of cores $|\mathbb{V}|$, because each core is coupled to a single router by Constraint 2. REGO aims to find the best population with the highest fitness in *TG*, the task graph of the HGC platform, from the chromosome set $\mathbb{C}_{set}$, whose elements are independently generated.

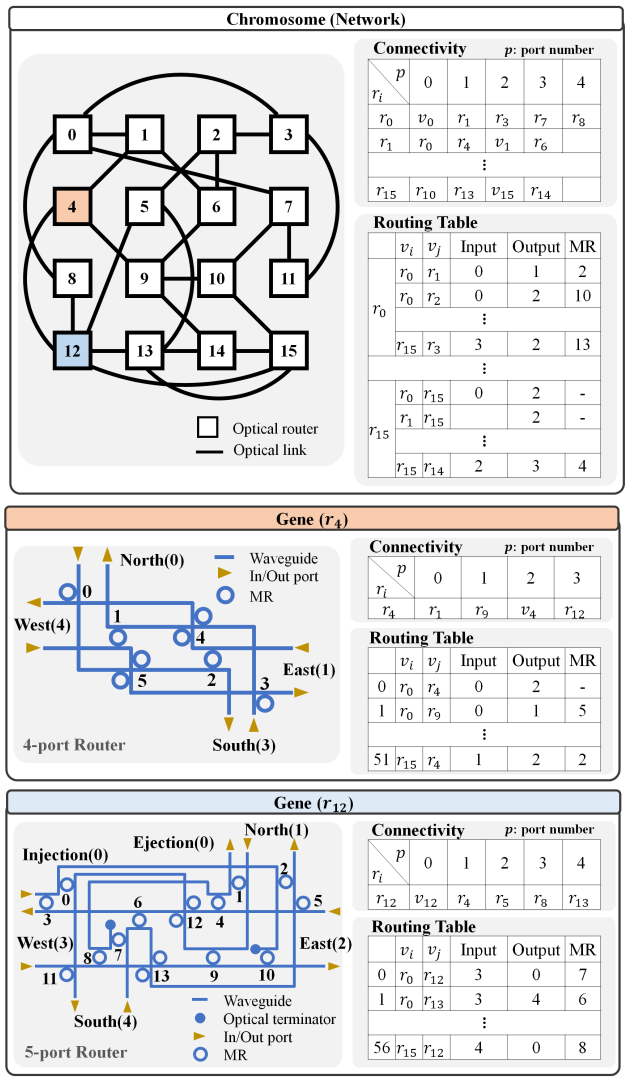

FIGURE 2. Example of chromosome configuration.

Because the worst-case OSNR of ONoC determines the laser power consumption while guaranteeing the required performance, it is necessary to set up a fitness function suitable for the ONoC environment. Therefore, the worst-case OSNR is an essential objective when assessing the fitness function [35].

As aforementioned, the MR heaters for tuning the resonance wavelength of the MR accounts for approximately 20 % of the total power consumption of the ONoC, and the number of MR heaters is identical to the number of MRs [34]. If a router with a large number of ports is used as a building block, the number of required MRs increases, whereas the hop count in the longest path decreases. This relationship indicates that a trade-off exists between the number of MRs and the worst-case OSNR. Therefore, we separate the worst-case OSNR and the number of MRs into different objectives in the fitness calculation regarding power minimization. We incorporate a fitness function $F(X, I)$ comprising the importance factor I_i and M objectives $O_i(X)$ for multi-objective optimization proposed in [36].

$$F(X, I) = \sum_{i=1}^{M} I_i[O_i(X)] \tag{1}$$

Because the HGC platform for deep learning requires high throughput with low power consumption, we focused on optimizing power and throughput. Therefore, power and throughput were considered as objectives of the fitness function in REGO. A modified fitness function of a chromosome $f(\mathbb{C})$ based on (1) is introduced in the REGO, which utilizes the throughput O_{thr}, worst-case OSNR O_{snr}, and number of MRs O_{mr} as objectives:

$$f(\mathbb{C}) = I_1 \cdot O_{thr} + I_2 \cdot O_{snr} + I_3 \cdot O_{mr}, \quad (I_1 + I_2 + I_3 = 1,\ 0 \le I_1, I_2, I_3 \le 1) \tag{2}$$

where I_i indicates the importance factor of i^{th} elements.

Adjusting the importance factors in the fitness function facilitates determining the optimized ONoC topology in terms of energy efficiency and throughput. Every element comprising the fitness function is scaled to an identical range for the uniform application of the importance factors.

B. INITIALIZATION PHASE

Population initialization is closely related to the convergence speed and quality of the final solution. Random initialization is commonly used to generate an initial population when the genetic information is not known in advance [37]. The GA guarantees randomness in the initialization by maximizing the diversity of genes in the chromosomes.

Algorithm 1 presents the process of the initialization phase of REGO. Each chromosome is sequentially initialized with a constraint that establishes a connected network. Because the number of available router ports is limited by the type of router permitted in the ONoC design, the process of randomly connecting routers is repeated until the number of ports of all routers reaches p_{min}, the minimum number of ports. After connecting all routers in $\mathbb{C}_i$, the path validity of each signal path *sp* is checked. If SP_i has an invalid path, adjusting *gl* to ensure network connectivity might harm the randomness of genetic data as well as increase computational complexity.

Therefore, in this case, the REGO abandons the entire router connection and repeats the above process.

Algorithm 1 Initialization Phase

In/Output Chromosome set $\mathbb{C}_{set}$
1: **for** $i = 1$ **to** $|\mathbb{C}_{set}|$ **do**
2: **while** !($\mathbb{C}_i$ constitutes connected network) **do**
3: remove all connectivity of $\mathbb{G}_i$ in $\mathbb{C}_i$
4: **for** $j = 1$ **to** $|\mathbb{R}|$ **do**
5: randomly select r_k of $\mathbb{R}$ in $\mathbb{C}_i$, $k \neq j$ where $nrp_k < p_{max}$ until $nrp_j < p_{min}$
6: update gl_j and gl_k
7: **if** ($p_{min} \leq nrp_k$ && $nrp_k \leq p_{max}$) **then**
8: update gt_j and gt_k
9: **end if**
10: **end for**
11: check path validity of sp_{jk} in $\mathbb{C}_i$ where $1 \leq j, k \leq |\mathbb{V}|$
12: **end while**
13: generate routing table RT_i of $\mathbb{C}_i$
14: sort $\mathbb{C}_i (\in \mathbb{C}_{set})$ in descending order of fitness value $f(\mathbb{C}_i)$
15: **end for**

C. EVOLUTION PHASE

The procedure of the evolution phase in REGO is depicted in Fig. 3. The selection, crossover, and mutation steps of the evolution phase are performed according to λ, ξ, and μ, respectively. In the selection step, chromosomes with high fitness are propagated to the next generation with selection probability p. In the crossover step, chromosomes with higher fitness than the previous generation are generated through the exchange of genetic data between chromosomes. In the mutation step, randomness is assigned to the chromosome set by transforming the genetic data in random range of the chromosomes.

1) SELECTION

The selection step in the REGO propagates chromosomes with high fitness values to the next generation. Although the fitness value of the chromosome is low, the dominant gene to elevate fitness value might be contained in the corresponding chromosome; thus, all chromosomes should be given an opportunity to be preserved for the next generation. In REGO, the chromosomes are initialized in a serial manner, and then sorted according to fitness values. Herein, the pre-sorted chromosomes in the initialization phase dramatically relieve the computational complexity of tournament selection.

Thus, the REGO uses tournament selection, which assigns ranks based on the fitness value and selects with a selection probability p_s. In the crossover and mutation of REGO, only the chromosome with modified fitness value needs to be sorted and thus the computational complexity is decreased compared to the initial sorting. Furthermore, tournament selection offers the benefit of reducing the enormous calculation time required to find the worst-case OSNR.

2) CROSSOVER

After completing the selection step, the chromosomes of the remaining $\mathbb{C}_{set}$ are selected for crossover according to

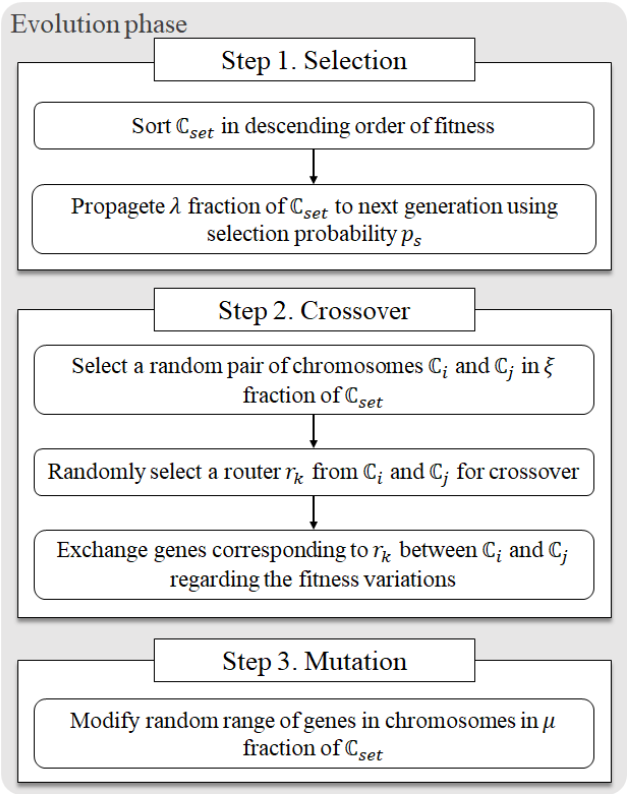

FIGURE 3. Evolution phase of REGO.

the crossover rate ξ. In the crossover step of the REGO, the genetic data gt and gl representing the type and connectivity of the optical router are exchanged between selected chromosomes. To comply with the basic constraints of REGO in Section 3.A, all chromosomes must satisfy the following three crossover conditions (COCs):

- COC 1. $\forall_{rt_{ij}} : rt_{ij} \neq \emptyset, i, j \in \{1, \cdots, |V|\}, i \neq j$
- COC 2. $\forall_{nrp_i} : nrp_i \leq p_{max}, i \in \{1, \cdots, |R|\}$
- COC 3. $\forall_{nrp_i} : nrp_i \geq p_{min}, i \in \{1, \cdots, |R|\}$

The REGO introduced a two-point crossover method for exchanging single router to avoid overlapping cases that violated the above COCs. If COC 1 is not satisfied after the crossover step, the exchanged router can be regarded as the dominant gene that determines the connectivity of the entire network. Forcing the crossover by connecting remaining isolated routers for COC 1 invalidates the effect of the crossover because most of the connectivity must be migrated from the previous generation. Therefore, when a case that violates COC 1 occurs, different gene or chromosome is newly selected for crossover in the REGO.

Chromosomes that violate COCs 2 or 3 frequently appear during the evolution phase iterations. Selecting only genes and chromosomes that fulfill all three COCs for crossover severely reduces the diversity of the chromosome set. Therefore, REGO should search for alternative routers that satisfy only COCs 2 and 3. The alternative routers must maintain the connection properties affecting on the fitness of the router originally intended to be connected.

Fig. 4 illustrates the architecture of a Cygnus router, which is a 5 × 5 optical non-blocking router, where the MR state is *ms* and the input-to-output ratio in the lookup table (LUT) of the signal power is $LUT[i][ms][s][j]$. The output signal power of the j^{th} port of the Cygnus router $P^{out}_{s,j}$ can be calculated using (3) according to $P^{in}_{s,i}$, the input signal power of the i^{th} port [[35]].

$$P^{out}_{s,j} = P^{in}_{s,i} \cdot LUT[i][ms][s][j], \quad i, j \in \{0, 1, \cdots, 4\} \tag{3}$$

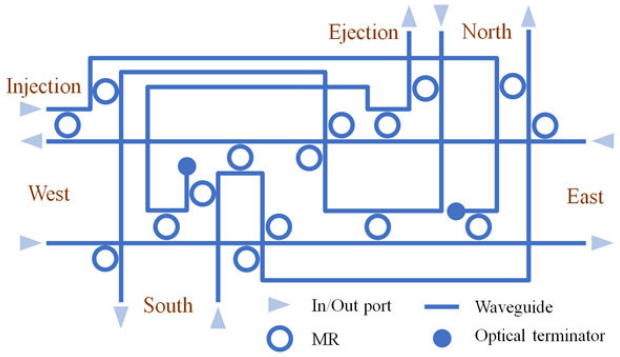

FIGURE 4. Organization of Cygnus router.

Similarly, when the MR state is *ms* and the noise power LUT is $LUT[i][ms][n][j]$, the output noise power of the j^{th} port of the Cygnus router is $P^{out}_{n,j}$ can be calculated using (4) according to $P^{in}_{n,i}$, which is the input noise power of the i^{th} port.

$$P^{out}_{n,j} = \sum_{k=0}^{n}((P^{in}_{s,k} + P^{in}_{n,k}) \cdot LUT[k][ms][n][j]), \quad i, j \in \{0, 1, \cdots, n\} \tag{4}$$

Assuming that the signal and noise power of every input port are commonly P^{in}_{s} and P^{in}_{n}, respectively, the OSNR of the j^{th} output port $OSNR_j$ is calculated using (5)

$$OSNR_j = \frac{LUT[i][ms][s][j]}{\sum_{k=0}^{n}(R[k][ms][n][j])(1 + P^{in}_{n}/P^{in}_{s})} \tag{5}$$

It should be noted that the term $P^{in}_{n}?P^{in}_{s}$ of the denominator in (5) is much less than 1 for guaranteeing signal reliability. Thus, the minimum OSNR, insertion loss, and crosstalk coefficient in the Cygnus router calculated using (5) are 13.44, −0.69 and −14.13 dB, respectively, which increased in proportion to the number of Cygnus routers on the routing path.

The packet delay P_{delay} ignoring data collision is expressed as (6), where OR_{bw}, OR_h, ER_{clk}, and ER_{pipe} indicate the link bandwidth, the number of hops of the routing path, the operating clock frequency, and the number of pipelines of the electrical router, respectively.

$$P_{delay} = \frac{1}{OR_{bw}} + \frac{OR_h \cdot ER_{pipe}}{ER_{clk}} \tag{6}$$

Because the packet delay increases as OR_h increases, the effect on the fitness function throughput perspective decreases. Consequently, the replaced router to avoid the COC violations must be placed adjacent to the router originally intended to be connected considering both the OSNR and throughput.

Algorithm 2 Crossover Step

In/Output Chromosome set $\mathbb{C}_{set}$
1: select chromosomes for crossover in $\mathbb{C}_{set}$ based on ξ
2: select a random pair $(\mathbb{C}_i, \mathbb{C}_j)$ of the selected chromosomes $(i, j \in |\mathbb{C}_{set}|)$
3: randomly select r_k of $\mathbb{R}$
4: $\mathbb{C}_{tmp\_i} = \mathbb{C}_i$, $\mathbb{C}_{tmp\_j} = \mathbb{C}_j$
5: **while** !(COC 1) **do**
6: randomly select r_k of $\mathbb{R}$
7: replace gl_k and gt_k in $\mathbb{C}_i$ to gl_k and gt_k in $\mathbb{C}_{tmp\_j}$
8: replace gl_k and gt_k in $\mathbb{C}_j$ to gl_k and gt_k in $\mathbb{C}_{tmp\_i}$
9: **if** gl_k violates the COCs 2 and 3 **then**
10: replace gl_k to the closest router from gl_k causing minimum fitness variation in RT
11: **end if**
12: check path validity of sp_{jk} in C_i and C_j where $1 \leq j, k \leq |\mathbb{V}|$
13: **end while**
14: **if** both $\mathbb{C}_i$ and $\mathbb{C}_j$ satisfy network connectivity **then**
15: update gt_k according to nrp_k
16: mark $\mathbb{C}_i$ and $\mathbb{C}_j$ to avoid redundant crossover
17: **else**
18: recover $\mathbb{C}_i$ and $\mathbb{C}_j$ based on $\mathbb{C}_{tmp\_i}$ and $\mathbb{C}_{tmp\_j}$
19: **end if**

Algorithm 2 describes the behavior of the crossover step in the REGO. The REGO based on two-point crossover randomly selects a single router that contains genetic information to be exchanged. Chromosomes $\mathbb{C}_i$ and $\mathbb{C}_j$ selected for crossover are stored as temporary variables for exchange and recovery. The router link gl_k violating the COC is replaced by a valid router link with regard to the fitness variation.

Fig. 5 shows an example of the crossover step in the REGO, where r_5 is selected as the target router to be exchanged between $\mathbb{C}_1$ and $\mathbb{C}_2$ when p_{min} is four and p_{max} is five in a 16-core ONoC. $\mathbb{C}_2$ can accept the genetic data of $\mathbb{C}_1$ without violating the COCs, whereas three violations of COCs 1 and 2

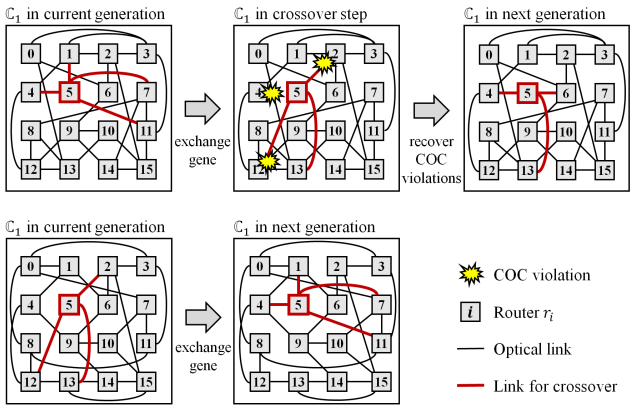

FIGURE 5. Example of crossover step.

occur in $\mathbb{C}_1$ in the crossover step. As gl_5 of $\mathbb{C}_1$ is replaced with gl_5 of $\mathbb{C}_2$, the number of ports of r_4 in $\mathbb{C}_1$, nrp_4, becomes three including the link with the core, which is less than p_{min}. In addition, nrp_2 and nrp_{12} become larger than p_{max} as they were equal to p_{max} before the crossover. Accordingly, r_4 with insufficient number of ports, is connected to the target router r_5 to satisfy the port constraint. The available port in r_5 intended to be assigned for connecting to r_{12} is reallocated to link r_4. Thus, COC 2 and COC 3 violations caused by r_4 and r_{12}, respectively, are simultaneously resolved. Finally, r_2 is replaced by r_6, which is adjacent to r_2, as revealed by the routing table RT. In this way, the crossover step of REGO reduces fitness variation by minimizing inevitable gene modification when recovering COC violations. Consequently, the convergence speed in evolution phase is accelerated by the crossover along with the tournament selection in REGO.

3) MUTATION

Mutation is a unique way of assigning additional randomness to the chromosome set, unlike selection and crossover, which depend on the diversity of the initial population [38]. In each chromosome of remaining $\mathbb{C}_{set}$ with regard to the mutation rate μ after the selection and crossover steps, an independent random range of genes in $\mathbb{G}$ is selected to be mutated. The REGO maximizes the randomness of chromosome set by randomly modifying the selected genes that include the type and connectivity of the optical router. Fig. 6 shows an example of a mutation in $\mathbb{C}_x$. Mutation starts by searching for link candidates to be added or deleted. REGO generates a mutated chromosome by randomly selecting some candidates. When the mutated $\mathbb{C}_x$ satisfies Constraint 2, $\mathbb{C}_x$ is replaced with a mutated chromosome. To comply the Constraint 2 of Section III.A, each mutation confirms the path validity of the chromosome containing the modified genes.

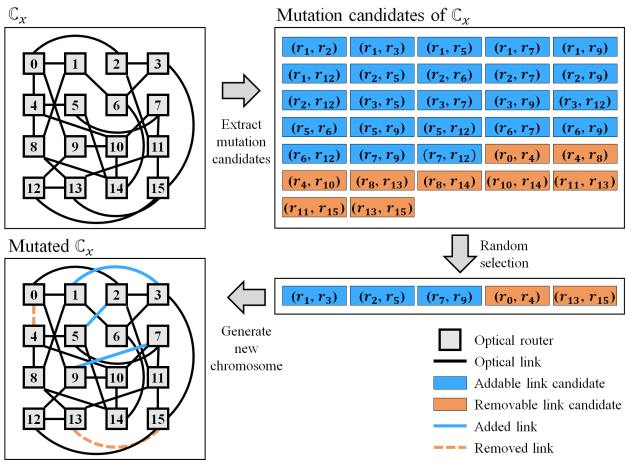

FIGURE 6. Example of mutation step.

IV. EVALUATION

A SystemC-based cycle-accurate simulator was built with ONoC parameters extracted through the linear optical device model (LODM) proposed in [12]. The visual geometry group (VGG), an academic group focused on computer vision at Oxford University, was dedicated to developing VGG-16 and VGG-19 16-layer and 19-layer deep convolutional networks, respectively [39]. We adopted the target application model of HGC as VGG-16 and VGG-19 [39], which are widely applied DNN models. Through the simulation, the throughput, latency, and energy efficiency of the derived ONoC topology and the convergence speed of the REGO were measured. To demonstrate the algorithmic complexity of the REGO, we compare the convergence speed of the REGO against the discrete binary PSO (BPSO)-based topology generation method and GA-based random mapping method that attempts to connect to the router randomly in the case of COC violation. We constructed the discrete BPSO-based topology generation method by extending the discrete BPSO-based core mapping method with the chaotic disturbance proposed in [16]. The extended discrete BPSO-based topology generation method uses the same fitness function, population size, and initial population as the REGO for a fair comparison. The velocity vector of the extended BPSO-based topology generation method includes the addition and deletion of connections between the routers, instead of swapping the position of the core. We assumed the maximum velocity of the extended discrete BPSO as three links, and applied chaotic disturbance when the movement of the location vector was inevitable owing to the network constraints of Section III.A. The random mapping method consists of the same GA parameters and processes as those of REGO, except for the responses of COC violation cases. We prepared 20 different initial populations to prevent inaccurate convergence speed derivation due to coincidence.

The throughput, latency, and energy consumption for irregular topology, regular topology mesh, and torus generated through REGO are compared to analyze the contribution of the REGO. Typically, the throughput in an NoC is defined as the amount of data transmitted per unit time. In this aspect, we assessed the throughput as the total data movement per unit time after all data transmission was completed in ONoC with the traffic of the target application. Latency per bit, considering data collision cases in real traffic, can be obtained as the reciprocal of throughput. Energy per bit is calculated by multiplying the power consumption according to the configuration of the ONoC and the latency per bit.

The specifications of the ONoC optical devices are listed in Table 3. In the mesh and torus, a Cygnus-based non-blocking optical router with four and five I/O ports was deployed [43]. A three-stage pipelined electrical router with an operating clock frequency of 1 GHz was assumed to control the optical layer.

According to [34], the MR heating power is a quarter of the laser power. Taking this relationship into account, we determine the importance factors I_1, I_2, and I_3 of (2) in Section III.A for 0.5, 0.4, and 0.1, respectively. In GA, ξ in the range of 60 %–90 % and μ in the range of 1 %—5 % can rapidly obtain the feasible solution [44]. To comply with this range, λ, ξ, and μ were set to 0.35, 0.6, and 0.05, respectively.

TABLE 3. Specifications of optical devices.

Parameter	Value	Reference
MR heating power	5 μW	[34]
Photodetector sensitivity	−20 dBm	[40]
Power margin of laser	13 dB	[41]
Transceiver bandwidth	10 Gb/s	[42]
Laser wall plug efficiency(L_e)	0.25	[41]

The OSNR is analyzed using optical elements with the parameters listed in Table 4 [45]–[49]. First, the router-level OSNR was calculated using the optical crossbar implemented by Verilog-AMS. Next, transport-level OSNR was analyzed through EWOSA [35], a high-speed OSNR analysis method.

TABLE 4. Loss and noise coefficients of optical elements.

Loss Parameter	Value	Noise Parameter	Value
Waveguide	−0.274 dB/cm	Optical Terminator	−50 dB
Waveguide Bending	−0.005 dB/90°	Waveguide Crossing Reflection	≈ 0
Waveguide Crossing	−0.04 dB	Waveguide Crossing	−40 dB
Crossing MR Through	−0.04 dB		
Crossing MR Drop	−0.5 dB		
Parallel MR Through	−0.005 dB	Parallel MR Through	−20 dB
Parallel MR Drop	−0.5 dB	Parallel MR Drop	−25 dB

The system configuration of the HGC system consisting of multicore CPUs, GPUs, and memory controllers (MCs) is presented in Table 5. Each GPU core consists of four unified shaders, and the last-level cache (LLC) is assumed to be an L2 cache shared between all CPUs and GPU cores.

A. CONVERGENCE SPEED ANALYSIS

Fig. 7 shows degree of the convergence speed of the REGO, GA-based random mapping method, and the discrete BPSO according to the iteration number of the evolution phase. In VGG-16, the fitness value of the REGO, GA-based random mapping method, and the discrete BPSO converged at average 3635^{th}, 5672^{nd}, and 1180^{th} iteration, respectively. In VGG-19, the fitness value of the REGO, GA-based random mapping method, and the discrete BPSO converged at average 4065^{th}, 4580^{th}, and 989^{th} iteration, respectively. The average convergence speed of the REGO was 1.56 and 1.13 times faster than GA-based random mapping method in the VGG-16 and the VGG-19, respectively. The convergence

TABLE 5. HGC platform architecture configurations.

Parameters	Configuration
GPU / shader core clock frequency	0.7 GHz / 1.4 GHZ
SIMT width	8
GPU Private L1 I cache size	64 KB
GPU Private L1 D cache size	64 KB
CPU clock frequency	2.5 GHz
CPU Private L1 I cache size	64 KB
CPU Private L1 D cache size	64 KB
Shared L2 cache size	1 MB per MC
DRAM	2 GB (DDR3-1600 MHz)

parameters	HGC-1		HGC-2	
	16 cores	32 cores	16 cores	32 cores
Number of CPU tile	2	4	1	2
Number of GPU tile	12	24	14	28
Number of MC	2	4	1	2

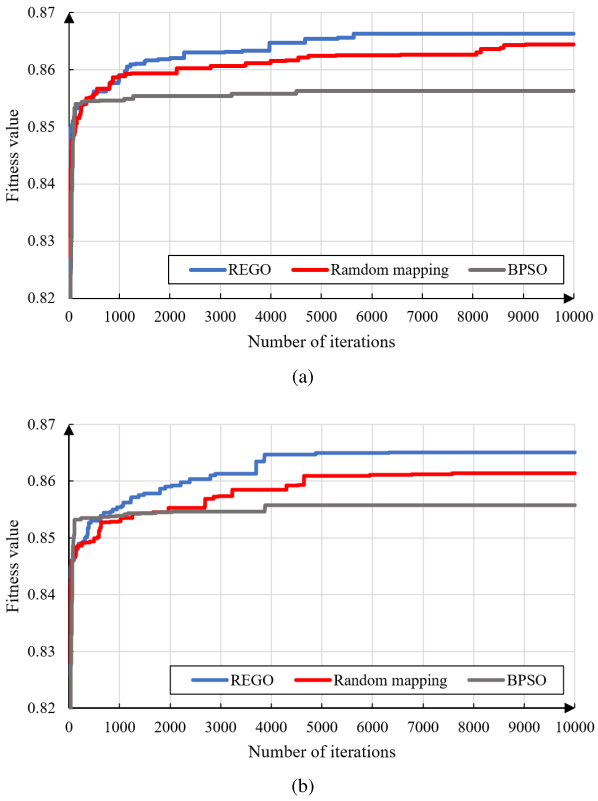

FIGURE 7. Fitness value comparison in terms of the number of iterations in evolution phase for (a) VGG-16, (b) VGG-19.

speed of discrete BPSO was faster than that of REGO in most cases. However, since discrete BPSO attempts to search only in the direction of the past and current best particles, a local best solution was defined along with a narrow search range. As a result, the average converged fitness of REGO was 0.003 and 0.011 higher than that of the random mapping method and discrete BPSO, respectively.

The average CPU runtime per iteration of REGO, GA-based random mapping method, and discrete BPSO was

TABLE 6. Comparison result of the irregular topology generated through REGO with regular topologies mesh and torus.

HGC-1					
		Throughput (GB/s)	OSNR (dB)	Number of MRs	Fitness value
VGG-16 16 cores	Mesh	20.520	10.141	224	0.823
	Torus	24.988	8.212	224	0.766
	REGO	29.807	8.005	192	0.839
VGG-19 16 cores	Mesh	19.779	10.188	224	0.827
	Torus	28.350	8.576	224	0.758
	REGO	30.169	8.160	208	0.841
VGG-16 32 cores	Mesh	19.836	7.120	448	0.803
	Torus	34.817	3.742	448	0.700
	REGO	43.103	6.883	400	0.844
VGG-19 32 cores	Mesh	23.537	7.330	448	0.799
	Torus	36.979	3.887	448	0.699
	REGO	37.206	6.132	400	0.843
HGC-2					
		Throughput (GB/s)	OSNR (dB)	Number of MRs	Fitness value
VGG-16 16 cores	Mesh	14.098	10.332	224	0.789
	Torus	17.640	8.147	224	0.718
	REGO	18.986	9.884	192	0.840
VGG-19 16 cores	Mesh	14.364	10.332	224	0.787
	Torus	17.401	7.947	224	0.717
	REGO	19.588	9.891	192	0.835
VGG-16 32 cores	Mesh	14.270	8.220	448	0.805
	Torus	21.700	3.601	448	0.678
	REGO	22.536	6.511	400	0.832
VGG-19 32 cores	Mesh	14.616	8.220	448	0.803
	Torus	22.096	3.831	448	0.681
	REGO	26.029	7.033	416	0.827

5.12, 5.23, and 10.25 secs, respectively. Discrete BPSO required approximately twice the CPU runtime compared to other GA-based two topology generation methods. Regarding the time complexity of the EWOSA TC_{EWOSA} and population size $N_{population}$, the time complexities of the REGO and discrete BPSO are expressed as $O(N_{population} \cdot (\xi + \mu) \cdot TC_{EWOSA})$ and $O(N_{population} \cdot TC_{EWOSA})$, respectively. Therefore, these results were caused by significant computational complexity of the discrete BPSO, which calculates the fitness for all particles. Moreover, the discrete BPSO was slowed down by additional searches for the router links that could be added or removed in all particles.

These results show that when deriving the best population, as shown in Fig. 7, crossing the dominant gene of the existing chromosome is more likely to generate a population with high fitness than a randomly generated gene, such as a mutation. These results reflect the fitness value variation of genetic data not covered by the random mapping method in the crossover step of REGO.

B. THROUGHPUT AND ENERGY EFFICIENCY ANALYSIS

Table 6 shows the results of comparing the irregular topology generated through REGO with regular topology mesh and torus topologies in terms of throughput, worst-case OSNR, number of MRs, and fitness. Increasing the number of links in the ONoC can improve the path diversity of the packet transmission. High path-diversity is beneficial to throughput, however, might involve the adversarial effect on OSNR caused by additional noise. For this reason, torus-based ONoC exhibited higher throughput and lower worst-case OSNR than mesh-based ONoC as shown in Table 6.

The irregular topology produced by the REGO showed a maximum throughput improvement of 117.30 % and 78.09 % in HGC-1 and HGC-2, respectively, compared to the conventional mesh topology (68.29 % and 51.76 % on average). In addition, compared to the torus topology, the throughput improvement of REGO of HGC-1 and HGC-2 was up to 23.80 % and 17.80 %, respectively, (12.53 % and 10.46 % on average). In the REGO, the dominant genes are exchanged while maintaining the existing properties considering the OSNR and throughput at the crossover step of the GA, therefore the throughput of ONoC is significantly increased.

The average fitness value of topology obtained by the REGO achieved 4.72 % and 22.77 % higher than that of mesh and torus, respectively. These results indicate that the irregular topology-based solution is required to optimize the multi-objectives desired in ONoC for the HGC platform.

Figs. 8 (a) and (b) show the normalized latencies of HGC-1 and HGC-2, respectively. The irregular topology obtained from the REGO had 60.03 % and 11.49 % lower average latencies comparing to the mesh and torus topologies, respectively. The irregular topology optimized for the application has a structural advantage over the existing regular topology in terms of latency.

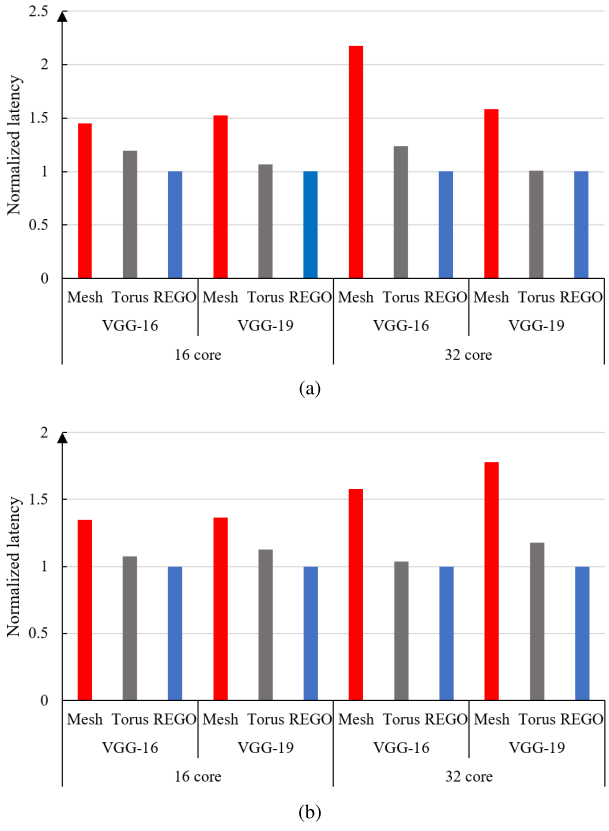

FIGURE 8. Normalized latency comparison results for VGG-16 and VGG-19 according to HGC platform architecture configurations in (a) HGC-1, (b) HGC-2.

Moreover, the lower latency of the irregular topology-based ONoC contributed to an increase in the energy efficiency.

Figs. 9 (a) and (b) show the average energy per bit for HGC-1 and HGC-2, respectively. In Table 6, in the 16-core network using the VGG-16 application, the mesh topology has approximately 2.14 dB higher worst-case OSNR than the topology obtained from the REGO. However, Fig. 9 presented that the energy per bit of the network generated from the REGO is 58.10 % lower than that of the mesh-based network because of the high latency with fixed number of MRs of the mesh-based ONoC.

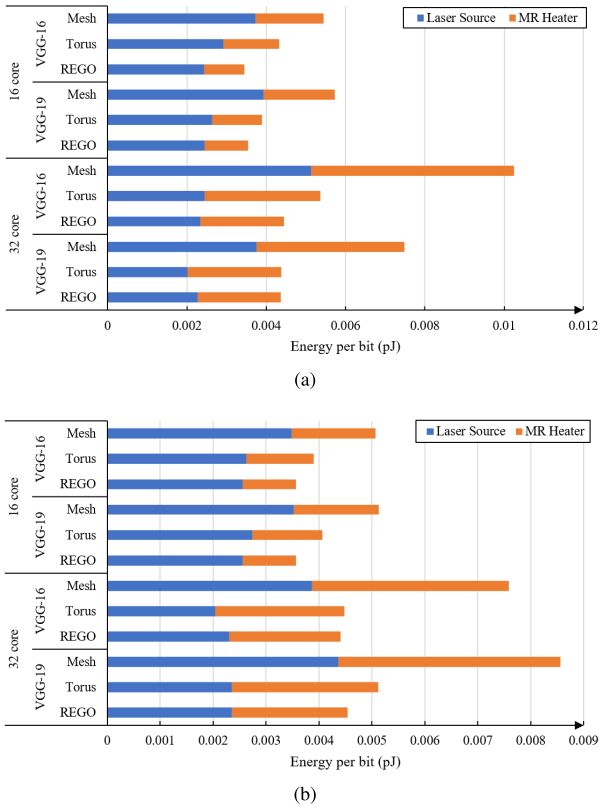

FIGURE 9. Energy per bit comparison results for VGG-16 and VGG-19 according to HGC platform architecture configurations in (a) HGC-1, (b) HGC-2.

The mesh topology has a relatively large MR heating power owing to the adaption of a fixed-structured router, and a 31.16 % bit latency difference significantly affects the energy per bit. Because this difference is noticeable in the 32-core network where the number of MRs increases, the solution obtained from REGO of 32 cores in HGC-1 has an average of 50.25 and 9.91% lower energy per bit compared to the mesh-based and torus-based networks, respectively. In conclusion, the ONoC implemented by the REGO can explore beyond the regular topology-based networks for DNN in terms of both energy efficiency and throughput.

V. CONCLUSION

In this paper, a GA-based REGO that enables the optimization of throughput and energy efficiency of ONoC required by the HGC was proposed. The ONoCs produced by the REGO achieved 63.29 % and 22.80 % higher throughput than mesh- and torus-based ONoCs, respectively, with high energy efficiency. Moreover, the crossover in REGO accelerated the convergence speed of fitness value by quick recovery of the design constraint violations. The REGO optimized the trade-off between the throughput and power consumption, considering OSNR variations based on the parameters of optical elements.

REFERENCES

[1] W. Choi, K. Duraisamy, R. G. Kim, J. R. Doppa, P. P. Pande, R. Marculescu, and D. Marculescu, "Hybrid network-on-chip architectures for accelerating deep learning kernels on heterogeneous manycore platforms," in *Proc. Int. Conf. Compil., Archit. Synth. Embedded Syst. (CASES)*, 2016, pp. 1–10.

[2] M. J. Schulte, M. Ignatowski, G. H. Loh, B. M. Beckmann, W. C. Brantley, S. Gurumurthi, N. Jayasena, I. Paul, S. K. Reinhardt, and G. Rodgers, "Achieving exascale capabilities through heterogeneous computing," *IEEE Micro*, vol. 35, no. 4, pp. 26–36, Jul. 2015.

[3] S. Fathololoumi, K. Nguyen, H. Mahalingam, M. Sakib, Z. Li, C. Seibert, M. Montazeri, J. Chen, J. K. Doylend, H. Jayatilleka, and C. Jan, "1.6Tbps silicon photonics integrated circuit for co-packaged optical-IO switch applications," in *Proc. Opt. Fiber Commun. Conf. (OFC)*, Mar. 2020, pp. 1–3.

[4] A. L. Porta, R. Dangel, D. Jubin, N. Meier, F. Horst, and B. J. Offrein, "Scalable and broadband silicon photonics chip to fiber optical interface using polymer waveguides," in *Proc. IEEE Opt. Interconnects Conf. (OI)*, Jun. 2017, pp. 13–14.

[5] P. Bakopoulos, K. Christodoulopoulos, G. Landi, M. Aziz, E. Zahavi, D. Gallico, R. Pitwon, K. Tokas, I. Patronas, M. Capitani, and C. Spatharakis, "NEPHELE: An end-to-end scalable and dynamically reconfigurable optical architecture for application-aware SDN cloud data centers," *IEEE Commun. Mag.*, vol. 56, no. 2, pp. 178–188, Feb. 2018.

[6] R. Meade, S. Ardalan, M. Davenport, J. Fini, C. Sun, M. Wade, A. Wright-Gladstein, and C. Zhang, "TeraPHY: A high-density electronic-photonic chiplet for optical I/O from a multi-chip module," in *Proc. Opt. Fiber Commun. Conf. (OFC)*, 2019, pp. 1–3.

[7] C. Li, M. Browning, P. V. Gratz, and S. Palermo, "LumiNOC: A power-efficient, high-performance, photonic network-on-chip," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 33, no. 6, pp. 826–838, Jun. 2014.

[8] J. Howard, S. Dighe, S. R. Vangal, G. Ruhl, N. Borkar, S. Jain, V. Erraguntla, M. Konow, M. Riepen, M. Gries, and G. Droege, "A 48-core IA-32 processor in 45 nm CMOS using on-die message-passing and DVFS for performance and power scaling," *IEEE J. Solid-State Circuits*, vol. 46, no. 1, pp. 173–183, Jan. 2011.

[9] D. Vantrease, R. Schreiber, M. Monchiero, M. McLaren, N. P. Jouppi, M. Fiorentino, A. Davis, N. Binkert, R. G. Beausoleil, and J. H. Ahn, "Corona: System implications of emerging nanophotonic technology," *ACM SIGARCH Comput. Archit. News*, vol. 36, no. 3, pp. 153–164, 2008.

[10] Y. Pan, J. Kim, and G. Memik, "FlexiShare: Channel sharing for an energy-efficient nanophotonic crossbar," in *Proc. HPCA 16th Int. Symp. High-Perform. Comput. Archit.*, Jan. 2010, pp. 1–12.

[11] A. Joshi, C. Batten, Y.-J. Kwon, S. Beamer, I. Shamim, K. Asanovic, and V. Stojanovic, "Silicon-photonic clos networks for global on-chip communication," in *Proc. 3rd ACM/IEEE Int. Symp. Netw. Chip*, May 2009, pp. 1–11.

[12] M. S. Kim, Y. W. Kim, and T. H. Han, "System-level signal analysis methodology for optical network-on-chip using linear model-based characterization," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 39, no. 10, pp. 2761–2771, Oct. 2020.

[13] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. IEEE Int. Conf. Neural Netw.*, vol. 4, Nov. 1995, pp. 1942–1948.

[14] J. H. Holland, "Genetic algorithms," *Sci. Amer.*, vol. 267, no. 1, p. 66—73, Jul. 1992.

[15] M. Imran, R. Hashim, and N. E. A. Khalid, "An overview of particle swarm optimization variants," *Procedia Eng.*, vol. 53, pp. 491–496, Mar. 2013.

[16] W. Lei and L. Xiang, "Energy- and latency-aware NoC mapping based on chaos discrete particle swarm optimization," in *Proc. Int. Conf. Commun. Mobile Comput.*, Apr. 2010, pp. 263–268.

[17] P. K. Sahu, T. Shah, K. Manna, and S. Chattopadhyay, "Application mapping onto mesh-based network-on-chip using discrete particle swarm optimization," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 22, no. 2, pp. 300–312, Feb. 2014.
[18] A. R. Fekr, A. Khademzadeh, M. Janidarmian, and V. S. Bokharaei, "Bandwidth/fault tolerance/contention aware application-specific NoC using PSO as a mapping generator," in *Proc. World Congr. Eng.*, Jun. 2010, pp. 247–252.
[19] K. Manna, P. Mukherjee, S. Chattopadhyay, and I. Sengupta, "Thermal-aware application mapping strategy for network-on-chip based system design," *IEEE Trans. Comput.*, vol. 67, no. 4, pp. 528–542, Apr. 2018.
[20] A. Thengade and R. Dondal, "Genetic algorithm–survey paper," in *Proc. MPGI Nat. Multi Conf.*, Jan. 2012, pp. 25–29.
[21] T. Lei and S. Kumar, "A two-step genetic algorithm for mapping task graphs to a network on chip architecture," in *Proc. Euromicro Symp. Digit. Syst. Design*, Sep. 2003, pp. 1–8.
[22] N. Choudhary, M. S. Gaur, V. Laxmi, and V. Singh, "Genetic algorithm based topology generation for application specific network-on-chip," in *Proc. IEEE Int. Symp. Circuits Syst.*, May 2010, pp. 3156–3159.
[23] G. Leary, K. Srinivasan, K. Mehta, and K. S. Chatha, "Design of network-on-chip architectures with a genetic algorithm-based technique," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 17, no. 5, pp. 674–687, May 2009.
[24] M. Sacanamboy-Franco, F. Bolaños-Martinez, Á. Bernal-Noreña, and R. Nieto-Londoño, "Genetic algorithm for task mapping in embedded systems on a hierarchical architecture based on wireless network on chip WiNoC," *Dyna*, vol. 84, no. 201, pp. 202–209, Jun. 2017.
[25] Y. Ye, X. Wu, J. Xu, W. Zhang, M. Nikdast, and X. Wang, "Holistic comparison of optical routers for chip multiprocessors," in *Proc. Anti-Counterfeiting, Secur., Identificat.*, Aug. 2012, pp. 1–6.
[26] Y. Xiao, S. Nazarian, and P. Bogdan, "Self-optimizing and self-programming computing systems: A combined compiler, complex networks, and machine learning approach," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 27, no. 6, pp. 1416–1427, Jun. 2019.
[27] T. Maqsood, K. Bilal, and S. A. Madani, "Congestion-aware core mapping for network-on-chip based systems using betweenness centrality," *Future Gener. Comput. Syst.*, vol. 82, pp. 459–471, May 2018.
[28] A. Alagarsamy, L. Gopalakrishnan, and S. Ko, "KBMA: A knowledge-based multi-objective application mapping approach for 3D NoC," *IET Comput. Digit. Techn.*, vol. 13, no. 4, pp. 324–334, Jul. 2019.
[29] G. Lai and X. Lin, "Floorplan-aware application-specific network-on-chip topology synthesis using genetic algorithm technique," *J. Supercomput.*, vol. 61, no. 3, pp. 418–437, Apr. 2011.
[30] G. E. Fen and W. U. Ning, "Genetic algorithm based mapping and routing approach for network on chip architectures," *Chin. J. Electron.*, vol. 19, no. 1, pp. 91–96, Jan. 2010.
[31] N. Venkataraman and R. Kumar, "Design and analysis of application specific network on chip for reliable custom topology," *Comput. Netw.*, vol. 158, pp. 69–76, Jul. 2019.
[32] A. B. Ahmed and A. B. Abdallah, "PHENIC: Silicon photonic 3D-network-on-chip architecture for high-performance heterogeneous many-core system-on-chip," in *Proc. 14th Int. Conf. Sci. Techn. Autom. Control Comput. Eng. (STA)*, Dec. 2013, pp. 508–516.
[33] S. Vanwinkle and A. K. Kodi, "SHARP: Shared heterogeneous architecture with reconfigurable photonic network-on-chip," *ACM J. Emerg. Technol. Comput. Syst. (JETC)*, vol. 14, no. 2, pp. 1–22, Jul. 2018.
[34] S. Werner, J. Navaridas, and M. Luján, "A survey on optical network-on-chip architectures," *ACM Comput. Surv.*, vol. 50, no. 6, pp. 1–37, Jan. 2018.
[35] Y. W. Kim, J. H. Lee, and T. H. Han, "Extended worst-case OSNR searching algorithm for optical network-on-chip using a semi-greedy heuristic with adaptive scan range," *IEEE Access*, vol. 8, pp. 125863–125873, 2020.
[36] X. Wu and Y. Zhu, "An optimization method for importance factors and beam weights based on genetic algorithms for radiotherapy treatment planning," *Phys. Med. Biol.*, vol. 46, no. 4, pp. 1085–1099, Apr. 2001.
[37] S. Rahnamayan, H. R. Tizhoosh, and M. M. A. Salama, "A novel population initialization method for accelerating evolutionary algorithms," *Comput. Math. with Appl.*, vol. 53, no. 10, pp. 1605–1614, May 2007.
[38] O. Kramer, "Genetic algorithm essentials," in *Studies in Computational Intelligence*. Berlin, Germany, 2017.
[39] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*. [Online]. Available: http://arxiv.org/abs/1409.1556
[40] F. E. Doany, B. G. Lee, S. Assefa, W. M. J. Green, M. Yang, C. L. Schow, C. V. Jahnes, S. Zhang, J. Singer, V. I. Kopp, J. A. Kash, and Y. A. Vlasov, "Multichannel high-bandwidth coupling of ultradense silicon photonic waveguide array to standard-pitch fiber array," *J. Lightw. Technol.*, vol. 29, no. 4, pp. 475–482, Feb. 2011.
[41] I. O'Connor and G. Nicolescu, *Integrated Optical Interconnect Architectures for Embedded Systems*. Berlin, Germany: Springer, 2012.
[42] K.-H. Lee, D. J. Shin, H.-C. Ji, K. W. Na, S. G. Kim, J. K. Bok, Y. S. You, S. S. Kim, I. S. Joe, S. D. Suh, J. H. Pyo, Y. H. Shin, K. H. Ha, Y. D. Park, and C. H. Chung, "10Gb/s silicon modulator based on bulk-silicon platform for DRAM optical interface," in *Proc. Opt. Fiber Commun. Conf./Nat. Fiber Optic Eng. Conf.*, Mar. 2011, pp. 1–3.
[43] R. Ji, L. Yang, L. Zhang, Y. Tian, J. Ding, H. Chen, Y. Lu, P. Zhou, and W. Zhu, "Microring-resonator-based four-port optical router for photonic networks-on-chip," *Opt. Exp.*, vol. 19, no. 20, pp. 18945–18955, Sep. 2011.
[44] T. Manning, R. D. Sleator, and P. Walsh, "Naturally selecting solutions: The use of genetic algorithms in bioinformatics," *Bioengineered*, vol. 4, no. 5, pp. 266–278, Sep. 2013.
[45] P. Dong, W. Qian, S. Liao, H. Liang, C. C. Kung, N. N. Feng, R. Shafiiha, J. Fong, D. Feng, A. V. Krishnamoorthy, and M. Asghari, "Low loss silicon waveguides for application of optical interconnects," in *Proc. IEEE Photon. Soc. Summer Topicals*, Jul. 2010, pp. 191–192.
[46] F. Xia, L. Sekaric, and Y. Vlasov, "Ultracompact optical buffers on a silicon chip," *Nature Photon.*, vol. 1, no. 1, pp. 65–71, 2007.
[47] W. Ding, D. Tang, Y. Liu, L. Chen, and X. Sun, "Compact and low crosstalk waveguide crossing using impedance matched metamaterial," *Appl. Phys. Lett.*, vol. 96, no. 11, 2010, Art. no. 111114.
[48] J. Chan, G. Hendry, K. Bergman, and L. P. Carloni, "Physical-layer modeling and system-level design of chip-scale photonic interconnection networks," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 30, no. 10, pp. 1507–1520, Oct. 2011.
[49] G.-R. Zhou, X. Li, and N.-N. Feng, "Design of deeply etched antireflective waveguide terminators," *IEEE J. Quantum Electron.*, vol. 39, no. 2, pp. 384–391, Feb. 2003.

YONG WOOK KIM (Student Member, IEEE) received the B.S. degree in electronic and electrical engineering from Sungkyunkwan University, Suwon, South Korea, in 2016, where he is currently pursuing the M.S. and Ph.D. degrees in electrical and computer engineering. His research interests include NoC, machine learning, and computer architecture.

SEO HONG CHOI (Student Member, IEEE) received the B.S. degree in electronic engineering from Soongsil University, Seoul, South Korea, in 2020. He is currently pursuing the M.S. and Ph.D. degrees in artificial intelligence with Sungkyunkwan University, Suwon, South Korea. His research interests include NoC, deep neural networks, and heterogeneous computer architecture.

TAE HEE HAN (Member, IEEE) received the B.S., M.S., and Ph.D. degrees in electrical engineering from Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea, in 1992, 1994, and 1999, respectively. From 1999 to 2006, he was with the Telecom Research and Development Center, Samsung Electronics, where he developed 3G wireless, mobile TV, and mobile WiMax handset chipsets. Since March 2008, he has been with Sungkyunkwan University, Suwon, South Korea, as a Professor. From 2011 to 2013, he was a full-time Advisor on system ICs with Korean Government. His current research interests include SoC architecture for artificial intelligence, emerging memory systems, and processing in memory.