Optical Switching and Networking 27 (2018) 50–57

Contents lists available at ScienceDirect

Optical Switching and Networking

journal homepage: www.elsevier.com/locate/osn

A novel IP-core mapping algorithm in reliable 3D optical network-on-chips

Lei Guo, Yifan Ge, Weigang Hou*, Pengxing Guo*, Qing Cai, Jingjing Wu

School of Computer Science and Engineering, Northeastern University, Shenyang 110819, China

ARTICLE INFO

Keywords:
3D ONoC
IP-core mapping
Reliability
Genetic and simulated annealing algorithm

ABSTRACT

The Optical Network-on-Chip (ONoC) is considered as a promising way to achieve high performance of multiprocessor systems, and it will be a 3-Dimensional (3D) architecture organized by a certain topology where optical routers are optically interconnected with each other. For the design of 3D ONoCs, the highly reliable IP-core mapping is a key problem of properly assigning IP cores onto optical routers for a given communication task, and it has two main challenges: reliability estimation and mapping scheme. As for reliability estimation, crosstalk noise and thermal sensitivity which severely influence Signal-Noise-Ratio (SNR) should be measured. In addition, although standard genetic algorithms have been widely utilized to solve the optimal mapping solution due to the superiority of simple process, there are some deficiencies such as premature convergence and inferior local searching. In this paper, the impact factors of ONoC reliability are measured by SNR and thermal models, and we also design a novel IP-core mapping algorithm called as CGSA (Cataclysm Genetic-based Simulated Annealing) based on proposed models. In CGSA, we integrate genetic with an improved simulated annealing algorithm assorted with cataclysm strategies, in order to speed up the searching process. Furthermore, to enhance the network reliability, CGSA is bound with the topology selection, i.e., CGSA generates the optimal mapping solution with the best matched 3D ONoC topology. Simulation results show that CGSA is effective on achieving the higher reliability than benchmarks.

1. Introduction

The integration of many hardware functions and software routines on a single silicon chip becomes the development trend of electronic systems [1]. However, the Network-on-Chip (NoC) component has suffered from severe performance bottleneck in terms of high power consumption, transmission delay, and low bandwidth provisioning, as predicted by the ITRS roadmap. Therefore, the Optical Network-on-Chip (ONoC) is proposed as promising communication architecture for the future multiprocessor systems [2–4].

To systematically solve the problem caused by the bus architecture, ONoCs transplant computer network technology into the design of chips based on optical interconnections. Silicon photonics has become one of the most promising photonic integration platforms, which also promotes the development of ONoCs. Most importantly, the optical interconnection of silicon photonics has incomparable advantages of high bandwidth provisioning as well as low power dissipation and transmission delay. On the other hand, the 3-Dimensional (3D) integrated circuit is an attractive solution to overcoming the barriers to decreasing an interconnection scale [5]. To combine the advantages of optical interconnection and 3D integration, researchers proposed the concept of 3D ONoCs, i.e., a 3D architecture organized by a certain topology where optical routers are optically interconnected with each other. Compared with traditional 2D structures, 3D ONoC further reduces the physical connection length between a pair of chip lines, thus leading to the shorter transmission distance, the smaller transmission delay, and the lower power consumption [2–4].

IP-core mapping is a key problem when we design 3D ONoCs, and it has a significant influence on the network performance. IP-core mapping refers to the process of properly assigning IP cores onto optical routers for a given communication task. In general, IP-core mapping mainly contains two steps, as shown in Fig. 1. In step 1, the mapping solution generates using a mapping algorithm, and then it will be evaluated according to an estimation model in step 2, and feedback to step 1 for the solution optimizing adjustment [6]. Repeat two steps until the upper limit. Obviously, there are two main challenges of IP-core mapping in 3D ONoCs: mapping scheme and estimation model.

From the aspect of mapping schemes, genetic algorithms (GAs) were widely used to find mapping solutions for NoCs [7–9], but they merely considered a predefined topology and a single optimization objective. The authors in [7] proposed a multi-objective GA-based algorithm that not only jointly optimized power consumption, transmission delay, and chip area but also combined the optimal mapping solution with the topology selection. However, this approach is

* Correspondence to: School of Computer Science and Engineering Northeastern University, P. O. Box 365, Shenyang 110819, China.
E-mail addresses: houweigang@cse.neu.edu.cn (W. Hou), starguo@stumail.neu.edu.cn (P. Guo).

http://dx.doi.org/10.1016/j.osn.2017.08.001
Received 25 May 2016; Received in revised form 17 July 2017; Accepted 8 August 2017
Available online 22 August 2017
1573-4277/

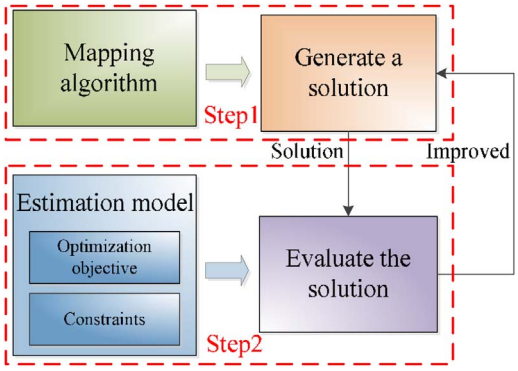

Fig. 1. IP-core mapping process.

confined in NoCs and cannot be extended into 3D ONoCs. Furthermore, standard genetic algorithms have two shortages when we solve an IP-core mapping problem: premature convergence phenomenon and inferior local searching.

Lots of researchers have designed accurate estimation models aimed at minimizing the power consumption and transmission delay for NoCs [10]. However, different from NoCs, the crosstalk noise and thermal sensitivity which severely influence Signal-Noise-Ratio (SNR) are intrinsic characteristics of photonic devices used by 3D ONoCs [11], and they should be taken into account for designing a novel reliability estimation model.

In this paper, we propose a methodology of finding the optimal IP-core mapping solution along with the best matched 3D ONoC topology, in order to ensure the high reliability. Our methodology includes three phases. Firstly, given a communication task graph, a novel mapping algorithm is designed to generate mapping solutions and their assorted 3D ONoC topology. Next, an original reliability model is proposed as the cost function of evaluating the reliability. Finally, we adjust the mapping solution according to evaluation results. Our contributions are summarized as follows.

1) To quickly search the mapping solution, a novel mapping algorithm based on the hybrid optimization strategy which combines improved GA and Simulated Annealing (SA) algorithm was proposed by us.
2) Used SA algorithm improves the ability of local exploration, and meanwhile, the cataclysm strategy applied to CGSA helps us jump out of the stagnation by generating new individuals.
3) An original reliability estimation model was designed to mathematically evaluate the 3D ONoC reliability from two aspects: crosstalk noise and thermal balancing.

The rest of this paper is organized as follows. Our reliability estimation model is in Section 2. Our Cataclysm Genetic-based Simulated Annealing (CGSA) optimization scheme is introduced in Section 3 for IP-core mapping and topology selection. Section 4 demonstrates simulation results of validating our work. Finally, we summarize the related work in Section 5 before concluding the paper in Section 6.

2. Reliability estimation model

Silicon waveguide bend/crossings and Micro-Resonator (MR)-based photonic switching elements are extensively used in Optical Routers (ORs), and their cumulative crosstalk noise degrades the SNR, thus leading to a high bit error rate, which directly menaces the reliability of 3D ONoCs. In addition, different communication requirements among IP cores cause an unbalanced thermal characteristic at various locations of the 3D ONoC. Thus, our novel reliability model is divided into two parts: SNR model and thermal model, in order to measure crosstalk noise and thermal balancing, respectively.

2.1. SNR analysis

We define the SNR as the ratio of the optical signal power P_S over the crosstalk power P_C, which can be written as Eq. (1).

$$SNR = 10 \log\left(\frac{P_S}{P_C}\right) \tag{1}$$

Our SNR model mainly has two parts: intra-OR SNR and inter-OR SNR. More specifically, we first calculate the intra-OR SNR based on the structure of ORs, and then statistically analyze inter-OR SNR in the packet and network levels.

2.1.1. Intra-OR SNR

The internal structure of ORs is shown in Fig. 2, which is built from waveguide bends, waveguide crossings and two types of 1×2 switching elements including Parallel Element (PE) and Crossing Element (CE) [9].

As shown in the left part of Fig. 2: a waveguide bend incurs a bending loss L_b, and if P_{in} is the power of the input optical signal, the

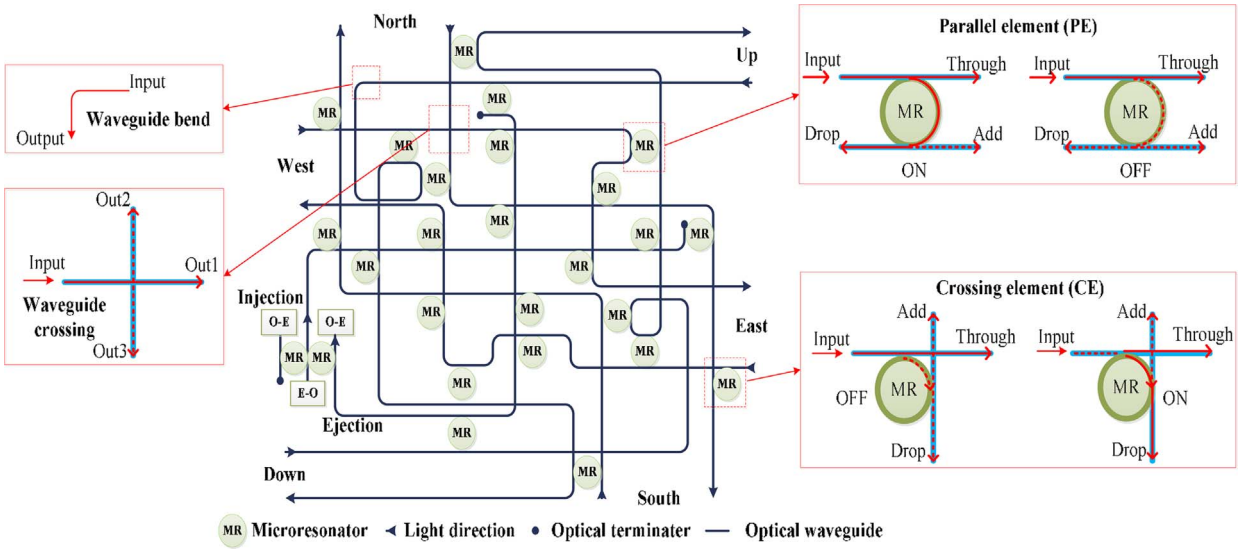

Fig. 2. Internal structure of optical routers.

output power $P_{out} = L_b \cdot P_{in}$; similarly, a waveguide crossing introduces a non-negligible crossing insertion loss L_c, thus, the output power at three ports are $P_{out1} = L_c \cdot P_{in}$ and $P_{out2} = P_{out3} = K_c \cdot P_{in}$, where K_c is the crosstalk coefficient per crossing.

As shown in the right part of Fig. 2: 1) as for the parallel element at OFF state, the optical signal passes through the MR to `Through' port, thus the corresponding output power at `Through' and `Drop' ports are $PE_{T,off} = L_{pe,off} \cdot P_{in}$ and $PE_{D,off} = K_{pe,off} \cdot P_{in}$; 2) while for the parallel element at ON state, the optical signal makes turns to `Drop port', thus the corresponding output power at `Drop' and `Through' ports are $PE_{D,on} = L_{pe,on} \cdot P_{in}$ and $PE_{T,on} = K_{pe,on} \cdot P_{in}$. Note that, the crosstalk on the `Add port' of PEs is negligibly small; 3) different from PEs, the CE has a waveguide crossing. Thus, the output power of different ports owned by the CE at OFF state $CE_{T,off} = L_{ce,off} \cdot P_{in}$, $CE_{D,off} = [K_{pe,off} + (L_{pe,off})^2 K_c] P_{in}$, $CE_{A,off} = K_c \cdot L_{pe,off} \cdot P_{in}$, respectively. When CE is at ON state, the output power of each port $CE_{D,on} = L_{ce,on} \cdot P_{in}$, $CE_{T,on} = K_{pe,on} \cdot L_c \cdot (1 + K_c \cdot L_{pe,on}) \cdot P_{in}$, $CE_{A,on} = K_{pe,on} \cdot K_c \cdot (1 + K_c \cdot L_{pe,on}) \cdot P_{in}$

Here, $L_{pe,off}$ and $K_{pe,off}$ are the power loss and crosstalk coefficient of the parallel element at OFF state, respectively; $L_{pe,on}$ and $K_{pe,on}$ are the power loss and crosstalk coefficient of the parallel element at ON state, respectively; $L_{ce,off}$ is the power loss of the crossing element at OFF state, and $L_{ce,on}$ is the power loss of the crossing element at ON state. The values of those parameters can be found in [12].

In this paper, we consider that the input optical signal power P_{in} at the injection port of different ORs is the same. The seven kinds of ports, including Injection/Ejection (IP Core), North, East, South, West, Up and Down in the middle part of Fig. 2, are numbered as 0, 1, 2, 3, 4, 5, 6, orderly. When there is an optical signal traveling from port i to port j within the OR located at the 3D coordinate (x, y, z), the corresponding intra-OR insertion loss is defined as $L_{i,j}(x, y, z)$, while the corresponding crosstalk noise introduced by that optical signal is $C_{i,j}(x, y, z) = \sum_{k \in [0,7]} P_k K_{i,j,k}$. Here, k is the port number of the current OR, and $K_{i,j,k}$ is the crosstalk coefficient introduced by the signal power P_k injected from the adjacent OR to the m^{th} port of the OR located at the 3D coordinate (x, y, z). Especially, $P_k = P_{in} \cdot L_{i_k,j_k}(x_k, y_k, z_k)$, and $L_{i_k,j_k}(x_k, y_k, z_k)$ is the insertion loss introduced by the signal traveling from the port i_k to the output port j_k within the adjacent OR located at the 3D coordinate (x_k, y_k, z_k).

As an example of the OR located at the 3D coordinate (2, 1, 0) in Fig. 3, there is the optical signal traveling from 0th port (IP Core) to 2th port (East), and this signal passes through 2 waveguide bends (Bend 1 and Bend 2), 3 waveguide crossings (from Crossing 1 to Crossing 3), 3 CEs at OFF state, 1 PE at OFF state and 1 CE at ON state. As a result, the insertion loss of that OR $L_{0,2}(2, 1, 0) = (L_b)^2 (L_c)^2 (L_{ce,off})^3 L_{pe,off} L_{ce,on}$ and the output power of that OR is $P_S = P_{in} L_{0,2}(2, 1, 0)$. The crosstalk injected into four ports of that OR is demonstrated by the green line in Fig. 3. Hence, the cumulative crosstalk noise of that OR can be calculated as $C_{0,2}(2, 1, 0) = P_5 K_{0,2,5} + P_1 K_{0,2,1} + P_4 K_{0,2,4} + P_2 K_{0,2,2}$. Wherein, the signal power $P_5 = P_{in} L_{0,6}$ since the insertion loss $L_{0,6}$ is introduced by the signal traveling from IP Core to the output Down port within the adjacent OR located at the 3D coordinate (2, 1, 1). Similarly, $P_1 = P_{in} L_{0,3}$, $P_4 = P_{in} L_{0,2}$, and $P_2 = P_{in} L_{0,4}$.

2.1.2. Inter-OR SNR

In the packet level, for the optical path traversing N ORs, the corresponding cumulative insertion loss and crosstalk noise are

$$L_{path} = \prod_{h=1}^{N} L_h, \tag{2}$$

$$C_{path} = \sum_{h=1}^{N} K_h L_h L_{h+1} \cdots L_N. \tag{3}$$

Here, L_h and K_h are the insertion loss and crosstalk noise of the hth OR, respectively, and they can be measured by the method mentioned in the last subsection.

Thus, we have the packet- and network-level SNRs in the following.

$$SNR_{path} = 10 \log\left(\frac{P_{in} L_{path}}{C_{path}}\right), \tag{4}$$

$$SNR_{total} = \sum_{P} SNR_{path}, \tag{5}$$

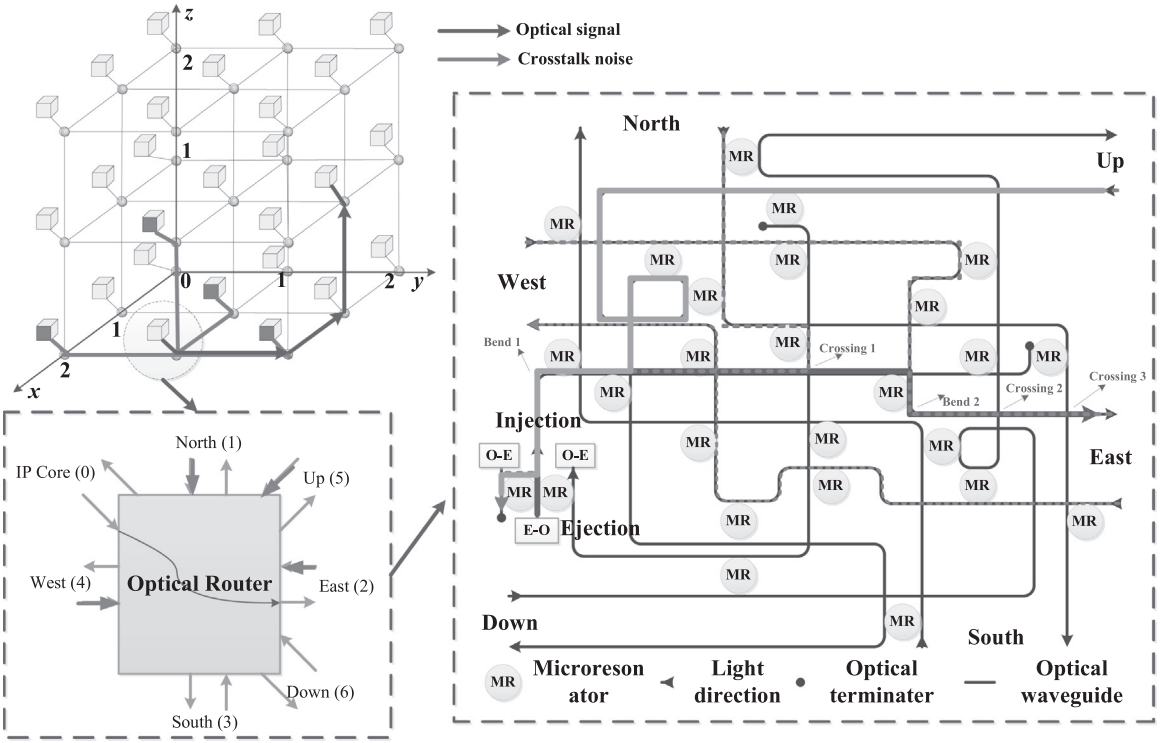

Fig. 3. An example of intra-OR SNR analysis.

where, P is the set of optical paths established in a 3D ONoC.

2.2. Thermal analysis

The thermal balancing in 3D ONoCs is closely related to both the data traffic and location of OR nodes. The closer that the data traffic of OR nodes is to the average network traffic, the more well-balanced thermal characteristic there is. Moreover, the thermal effect of the hotspot at the network center goes far beyond edge regions. Hence, our thermal model can be written as:

$$Ter = \frac{1}{m}\sum_{i=1}^{m}\left(\left|traf_{ni} - \frac{\sum_{i=1}^{m} traf_{ni}}{m}\right| \times e^{-\beta \cdot dis(n_i, n_c)}\right) \quad (6)$$

where, m is the total number of OR nodes carrying data traffic; $traf_{ni}$ is the data traffic of OR node n_i; $dis(n_i, n_c)$ is the distance between the OR node n_i and the OR node n_c located at the network center, i.e., $dis(n_i, n_c) = \sqrt{(x_i - x_c)^2 + (y_i - y_c)^2 + (z_i - z_c)^2}$; $\beta (0 \le \beta \le 1)$ is a constant of the distance impact factor.

2.3. Estimation model

Based on the above analysis, we define the following comprehensive estimation model as our optimization objective function.

$$\min\left\{f = \frac{(Ter)^{\alpha_B}}{(SNR_{total})^{\alpha_S}}\right\} \quad (7)$$

Where, α_B and α_S are the weight factors of thermal balancing and SNR, respectively, and they could be used to control the optimization process.

3. IP-core mapping scheme

IP-core mapping refers to the process of properly assigning IP cores onto ORs for a given communication task. Especially, the graph theory can be utilized to represent the communication task and the 3D ONoC topology deployed with ORs. Thus, we have the following definitions.

Definition 1:. The communication task graph $G(V, E)$ is a directed graph with $|V|$ vertexes and $|E|$ inter-core communication links. Each vertex $v_i \in V$ represents the IP core, while $e_{i,j} \in E$ is the directed virtual link corresponding to the communication trace from the IP core v_i to another IP core v_j. Each inter-core communication link is associated with the communication bandwidth requirement measured in MB/s, i.e., the traffic load. An illustration of that graph is part 1 of Fig. 4.

Definition 2:. The undirected topology can be denoted as one graph $G(N, L)$ that has $|N|$ nodes and $|L|$ edges. Each node $n_i \in N$ represents the OR, while each edge $l_i \in L$ represents the inter-OR waveguide between ORs n_i and n_j. Note that, Each OR has only one IP core, i.e., only one IP core can be mapped onto an OR. $|N| \ge |V|$. In this paper, we consider two kinds of topology candidates, 3D Torus and 3D Mesh, as shown in part 2 of Fig. 4.

Definition 3:. An inter-core communication link has its own communication trace that includes a sequence of inter-OR communication links.

Based on the aforementioned definitions, our IP-core mapping process can be generally described as follows. Given the communication task $G(V, E)$ (e.g., part 1 of Fig. 4) and S topology candidates $\{G_1(N_1, L_1), G_2(N_2, L_2), ...,G_S(N_s, L_s)\}$ (e.g., part 2 of Fig. 4), we first initialize p chromosomes as the first generation (e.g., part 3 of Fig. 4). Each chromosome is one candidate mapping solution that includes ($|N|$ + 1) elements. The first element is the ID of the selected topology, while the other $|N|$ elements represent the mapping relationship between OR and IP core. For example, the first element of the first chromosome shown in part 3 of Fig. 4 is 1, which denotes 3D Torus topology is the mapping solution. Also, the fifth element of the first chromosome shown in part 3 of Fig. 4 is 3, which means that the IP core 3 in part 1 of Fig. 4 has been mapped onto the OR 4 in part 2 of Fig. 4.

Next, the optimal chromosome (mapping solution) $\phi[v_i \in G(V, E) \rightarrow n_i \in G_*(N_*, L_*)]$ which has the minimal value of the estimation model in Eq. (7), i.e., maximizing the 3D ONoC reliability, will be found. Here, the optimal topology $G_*(N_*, L_*)$ is also selected from S candidates.

As previously mentioned, standard genetic algorithms have two shortages when we solve an IP-core mapping problem: premature convergence phenomenon and inferior local searching. To mitigate these shortcomings, we propose a novel IP-core mapping scheme Cataclysm Genetic-based Simulated Annealing (CGSA), in order to combine standard genetic algorithm with simulated annealing strategy which improves the ability of local exploration. Besides, Cataclysm strategy is applied to CGSA so that we can jump out of the stagnation by generating new individuals. The pseudo code of CGSA is in Algorithm 1, and an example of CGSA process is shown in Fig. 4. According to the given communication task and topology information, CGSA first initializes p chromosomes (i.e., mapping solutions) as the first generation. Evaluate p chromosomes in terms of the value owned by the estimation model in Eq. (7) (cost value in short), and choose the superior mapping solutions as the parent chromosomes for the genetic manipulation, as shown in lines 7–8. Here, the crossover and mutation are standard genetic operations of generating new mapping solutions from parent chromosomes. Evaluate the newly generated chromosomes in terms of the cost value, and then choose T superior chromosomes for simulated annealing. The output of the simulated annealing is denoted as m_{best} which is the optimal solution of the current iteration/generation, and it has the corresponding cost value $f(m_{best})$. We update m_{best} and its cost value at each iteration. In lines 10–15, if $f(m_{best})$ keeps unchanged within 5 iterations, the stagnation will be thought to occur, and the cataclysm strategy will be executed to help us jump out of this stagnation. The important parameters of Algorithm 1 are listed in Table 1. In the following, we describe the important operations of CGSA in detail.

Algorithm 1. CGSA algorithm

1. **Input:** $G(V, E)$, $\{G_1(N_1, L_1), G_2(N_2, L_2), ...,G_S(N_s, L_s)\}$, i=0; n=0.
2. **Output:** Mapping solution m_{best} with the corresponding optimal cost value $f(m_{best})$ calculated by Eq. (7).
3. Initialize the first Generation(p);
4. Calculate and sort the cost values of all chromosomes;
5. Choose the parent chromosomes using the roulette selection;
6. **While** $i < Maxiteration$ **do**
7. Crossover; Mutation;
8. Calculate and sort the cost value of all chromosomes;
9. Choose T superior chromosomes for simulated annealing;
10. **If** $f_i^* = f_{i-5}^*$ **then**
11. **While** $n < n_{cs}$ **do**
12. Execute cataclysm strategy;
13. n++;
14. **End While**
15. **End If**
16. i++;
17. **End While**
18. **Return** m_{best} and $f(m_{best})$.

3.1. Chromosome initialization

We initialize the first generation by randomly mapping every IP core in the communication task graph onto one of ORs in a 3D ONoC topology candidate. Then, p chromosomes (i.e., mapping solutions) including their corresponding topology ID are initialized. Thus, in

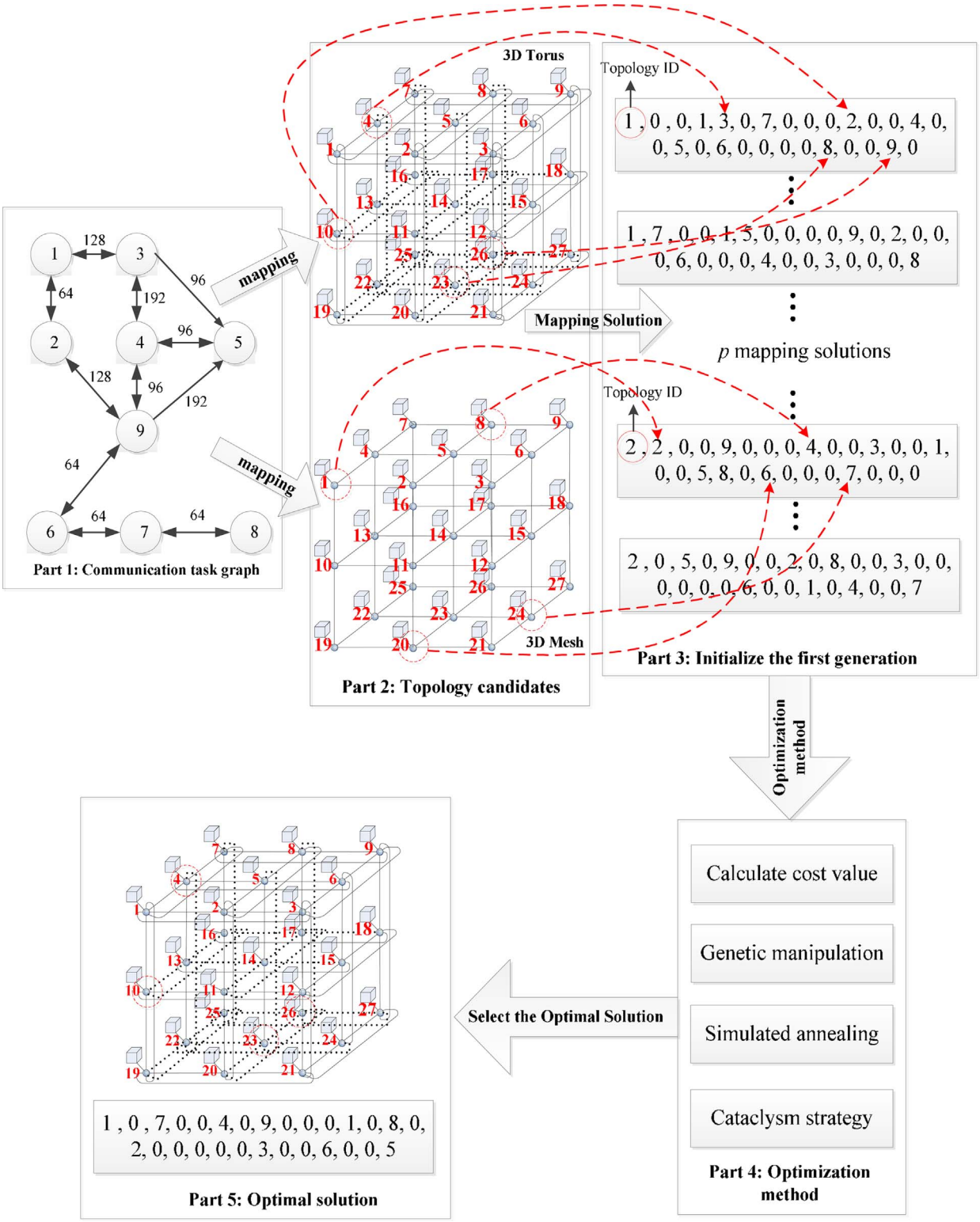

Fig. 4. An example of CGSA process.

essence, each chromosome is an integer vector with the length of ($|N| + 1$). As shown in Part 3 of Fig. 4, the first number in the vector is the ID of the selected topology, while the following $|N|$ numbers record the mapping relationship between IP core and OR. For example, in the vector of the first chromosome shown in the Part 3 of Fig. 4, the fifth number (i.e., '3') is the index of the IP core that has been mapped onto the OR indexed by '4' in the 3D Torus ONoC topology, while the number '0' means there are no IP cores mapped onto the current OR.

3.2. Simulated annealing

Simulated annealing is utilized for the further searching of the optimal one from several superior chromosomes. To enhance the

Table 1
Parameters in Algorithm 1.

p: The number of chromosomes initialized in each generation. p is set to 100 in simulations.
Maxiteration: The total number of iterations. *Maxiteration* is set to 200 in simulations.
T: The number of chromosomes selected for simulated annealing. T is set to 5 in simulations.
n_{cs}: The total number of cataclysm strategies. n_{cs} =16 in simulations.

Table 2
Parameters in Algorithm 2.

T_g: Temperature. T_g is set to 100 in simulations.
T_{min}: Ending temperature for the termination of simulated annealing. T_{min} is 0.1 in simulations.
U: Number of new chromosomes functioned as candidate mapping solutions generated from the multiple neighbors of the current chromosome. U=5 in simulations.
r: Temperature impact factor. r is set to 0.73 in simulations.

capability of local searching, identify the standard simulated annealing algorithm into the multi-neighbor simulated annealing that generates several candidate mapping solutions from the multiple neighbors of the current chromosome and then selects the optimal one. The pseudo code of the improved simulated annealing is shown in **Algorithm 2**. During the process of the improved simulated annealing, each iteration/generation has its own temperature T_g that decreases with the increasing number of iterations. Given a certain temperature, U new chromosomes are generated from the current chromosome x_i using the crossover function. Calculate the cost value of all U new chromosomes according to Eq. (7) and save the chromosome with the minimal cost value $f(x^*)$ as x^*. Finally, as shown in lines 10–16 of Algorithm 2, choose the better one from x^* and x_i to renew x_{best} and f_{best}. The important parameters in Algorithm 2 are listed in Table 2.

Algorithm 2. Multi-neighbor simulated annealing

1. **Input:** The current chromosome x_i.
2. **Output**: The chromosome x_{best} with the minimal cost value f_{best}.
3. **While** $T_g > T_{\min}$ **do**
4. **While** k < U **do**
5. Generate x_k with the crossover function;
6. Calculate the cost value $f(x_k)$ of x_k;
7. **End While**
8. $f(x^*) = \min\{f(x_1), f(x_2), \ldots, f(x_U)\}$;
9. $\Delta = f(x^*) - f(x_i)$;
10. **If** $\Delta \le 0$ **then**
11. $f_{best} \leftarrow f(x^*)$;
12. **Else**
13. **If** $\exp(\Delta/T_g) > random(0, 1)$ **then**
14. $f_{best} \leftarrow f(x^*)$;
15. **End If**
16. **End If**
17. $T_g = r \times T_g$;
18. $g++$;
19. **End While**

3.3. Cataclysm strategy

The cataclysm strategy reinitialized 10% chromosomes each of which previously has a poor cost value, in order to enhance the population diversity and expand the searching scope. Meanwhile, it keeps 90% high-quality chromosomes so that we can maintain the stabilization. Repeat the Cataclysm strategy until $n = n_{cs}$, and finally output the optimal mapping solution. The cataclysm strategy adds new chromosomes to expand the searching scope without changing the evolutionary consequence of the original population, thus promoting the efficiency.

4. Simulation results and discussions

In this section, we first introduce simulation settings and benchmarks. The comparative results between our CGSA and benchmarks are then demonstrated in terms of performance metrics. The simulation is performed by an open source NoC simulator, Noxim [13]. In our simulation, the real-time arriving of communication tasks is set artificially in Noxim because we cannot test a real ONoC chip. In other words, the communication task graph used here is randomly generated with the given number of IP cores and traffic load. Since Noxim is a simulator for 2D electrical NoCs, the simulator is modified by us to adapt to 3D ONoC topologies. The scale of candidate topologies (3D Mesh and 3D Torus) is $5 \times 5 \times 5$ where there are totally 125 ORs and 300 inter-OR waveguides. To demonstrate the superiority, we compare the performances among our CGSA, the standard Genetic Algorithm (GA) [9] and the standard Simulated Annealing (SA). Note that, the second benchmark SA means that only SA is utilized solve the IP-core mapping in 3D ONoCs, which can help us check whether the combination of GA and SA (i.e., our CGSA algorithm) performs better than the single SA or not. For the fairness, three algorithms use the same estimation model given in Eq. (7), and they are applied to 3D ONoCs.

First of all, the selection of initial parameters is very important for the algorithm to get the optimal solution. Correspondingly, the massive simulations can help us to find proper settings for some parameters. Especially for different crossover probabilities, we simulate our method under various kinds of communication tasks with 25 IP cores, 56 IP cores and 81 IP cores, as shown in Fig. 5. Note that, the objective function value determined by Eq. (7) under a certain crossover probability is the average of 100 times simulations. We can see that the increment of the crossover probability does not have a large effect on varying the objective function value, but when the crossover probability is around 0.8, the new chromosomes generated by the crossover function can make the objective function value become the lowest.

Fig. 6 shows the convergence curve of three algorithms under different numbers of iterations. From which we can see that compared to standard GA and SA, CGSA shows the superior at 40th iteration and it becomes more and more obvious with the increasing of iterations. The standard GA and SA start to converge at the 70th iteration and traps into a local optimum. Instead, CGSA mitigates the problem above and starts to converge at the around 80th iteration with the help of the cataclysm strategy. In addition, the mapping solution of GA, SA and

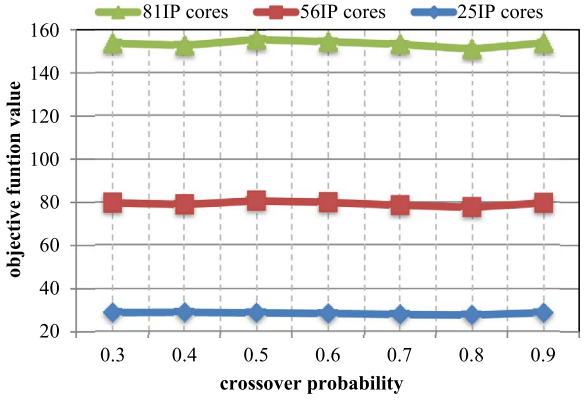

Fig. 5. Objective function value vs. crossover probability.

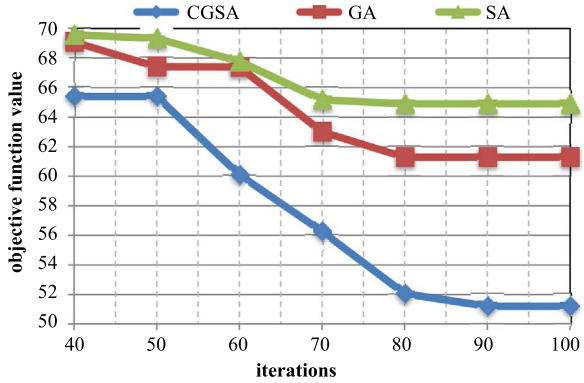

Fig. 6. Objective function value vs. number of iterations.

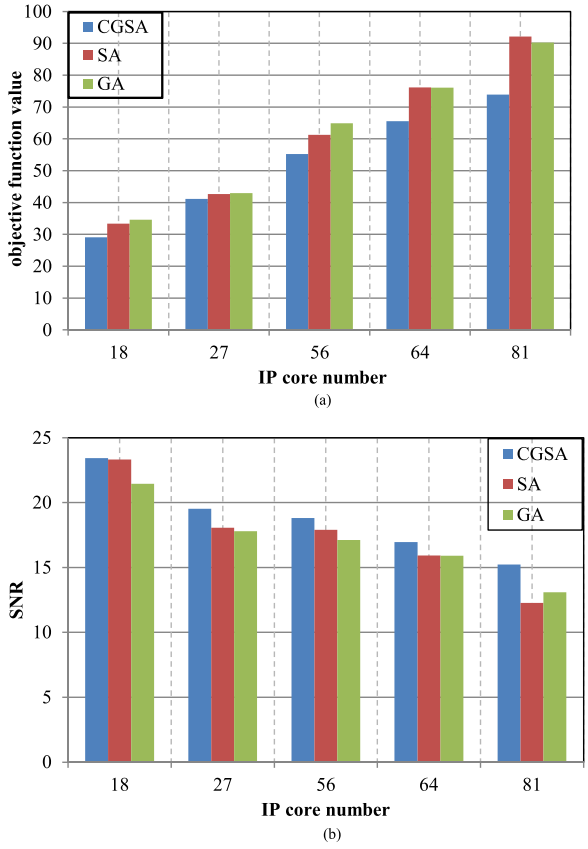

Fig. 7. Comparative results with different number of IP cores.

CGSA tend to be stable when the number of iterations reaches a certain threshold (e.g., 80), hence, we determined the *Maxiteration* as 100.

Fig. 7(a) and (b) respectively show the objective function value and SNR under different communication tasks with various amounts of IP cores. We can see that as the number of IP cores increases, the objective function value of three algorithms has a rising trend while SNR has an inverse tendency. CGSA always keeps good results compared to the benchmarks, and the advantage of CGSA become increasingly obvious as the number of IP cores rises. More specifically, the objection function value generated by CGSA is on average 19.8% lower than that generated by SA and 18.1% lower than that generated by GA, meanwhile, the SNR generated by CGSA is improved up to 24.1% and 16.3% compared with SA and GA, respectively. Note that, under the communication task with a small IP core number, SA achieves the better performance (the lower objective function value) than GA, contrary to that, GAs performed the better than SA. This is because that compared to GA, SA gains the better mapping solution

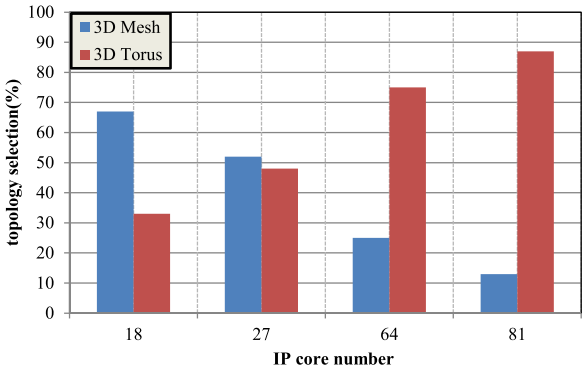

Fig. 8. Comparison of the selection probability of candidate topologies.

regardless of the solution solving speed, but GA has the large solution space which means the more choices of finding the optimal solution. CGSA combines the advantages of GA and SA, thus improving the local searching ability, avoiding the premature phenomena, and finally achieving the best performance (the lowest objective function value).

The comparison of the selection probability among different candidate topologies is conducted under various communication tasks in Fig. 8. The simulation results show that the 3D Mesh has the better performance in the cases of the small network scale and low traffic load, however, 3D Torus gradually demonstrates its advantages with the increment of IP core number and traffic load. The reason of this is that the added links in 3D Torus lead to the high crosstalk in the case of the small network scale and low traffic load, and meanwhile, it relieves the network congestion with the increment of IP core number and traffic load. Besides, 3D Torus has the better symmetry and extensibility compared to 3D Mesh.

5. Related work

Currently, there has been a list of works tailored to solving the IP-core mapping problem, and most of them are performed in 2D NoCs where energy consumption and transmission delay are both significant issues. While for 3D ONoCs, few innovative types of research can be found to improve the network reliability.

On the one hand, some existing mapping algorithms focus on the single optimization objective [14–18]. For example, by using the traditional genetic algorithm, the authors in [14] designed a mathematical model of analyzing the end-to-end transmission delay. Assorted with the cost function, the proposed algorithm could help the authors determine the optimal mapping solution. In [15], when there was a big communication volume, to improve the algorithm efficiency, the authors reduced the size of the search space using the task binding where communication tasks were sorted according to the corresponding volume, thus saving the energy consumption and facilitating the network performance improvement. In addition, an identified artificial swarm algorithm was proposed by [16,17] to solve the problem about early entering into stagnation during the iteration process. The simulation results demonstrated that this algorithm was able to obtain the better mapping solution with a more quicker convergence speed. Finally, a two-step genetic algorithm for mapping task graphs was proposed for a NoC architecture in [18]. However, the aforementioned related works focus on the single optimization objective in NoCs instead of 3D ONoCs which have their own features and requirements.

On the other hand, there were also some works focusing on designing multi-objective mapping solutions [19–21]. Among which, the authors in [20] proposed the mapping algorithm using chaotic discrete particle swarm optimizing the energy consumption and transmission delay in 2D mesh NoCs. This algorithm also introduced inertia item to avoid premature phenomenon. To achieve the good

balance between search space size and local optimization, the repulsion operator was utilized to keep the solution diversity. In [21], a mapping approach was designed to reduce the generation probability of the hot spot, thus prolonging the chip life, with the help of a dynamic genetic algorithm that could find a good trade-off between the processor load and network communication volume. However, the aforementioned multi-objective mapping solutions mainly optimized the energy consumption and transmission delay instead of thermal balancing which is one of the important topics to solve for 3D ONoCs.

6. Conclusion

In this paper, we have proposed a new method CGSA based on improved GA and SA to solve the IP-core mapping and topology selection problems in highly-reliable 3D ONoCs. Meanwhile, a novel estimation model has been designed to measure the network reliability with the consideration of the crosstalk noise and thermal sensitivity. Compared with the standard GA and SA, our approach has proved to be quite effective to find the optimal mapping solution with the most appropriate topology.

Acknowledgements

This work is supported by the National Nature Science Foundation of China under Grant 61401082, in part by the General Armament Department and Ministry of Education United Fund under Grant 6141A0224-003, and in part by the Fundamental Research Funds for the Central Universities under Grant N161604004 and Grant N161608001.

References

[1] M. Briere, E. Drouard, F. Mieyeville, et al., Heterogeneous modeling of an optical network-on-chip with system C, Proc. RSP (2005) 10–16.
[2] Q. Cai, W. Hou, C. Yu, et al., Design and OPNET implementation of routing algorithm in 3D optical network on chip, Proc.. ICCC (2014) 112–115.
[3] W. Hou, L. Guo, Q. Cai, et al., 3D Torus ONoC: topology design, router modeling and routing algorithms (Invited talk), Proc. ICOCN (2014) 1–4.
[4] P. Guo, W. Hou, L. Guo, et al., Reliable routing in 3D optical network-on-chip based on fault node reuse, Proc. RNDM (2015) 92–98.
[5] H. Ying, T. Hollstein, K. Hofmann, Fast and optimized task allocation method for low vertical link density 3-dimensional networks-on-chip based many core systems, Proc. DATE (2013) 1777–1782.
[6] X. Jin, N. Guan, Q. Deng, et al., Memory access aware mapping for networks-on-chip, Proc. RTCSA (2011) 339–348.
[7] A. Amorgan, H. Elmiligi, M.W. Elkharashi, et al., Unified multi-objective mapping and architecture customization of networks-on-chip, Comput. Digit. Tech. IET (2013) 282–293.
[8] V.A. Palaniveloo, J.A. Ambrose, Improving GA-based NoC mapping algorithms using a formal model, Proc. ISVLSI (2014) 344–349.
[9] X. Jiang, T. Watanabe, An efficient 3D NoC synthesis by using genetic algorithms, Proc. TENCON (2010) 1207–1212.
[10] J. Sepúlveda, M. Strum, A multi-objective approach for multi-application NoC mapping, Proc. LASCAS (2011) 1–4.
[11] Y. Ye, J. Xu, B. Huang, et al., 3-D mesh-based optical network-on-chip for multiprocessor system-on-chip, IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst. (2013) 584–596.
[12] Y. Xie, M. Nikdast, J. Xu, et al., Formal worst-case analysis of crosstalk noise in mesh-based optical networks on chip, IEEE Trans. VLSI Syst. (2013) 1823–1836.
[13] F. Fazzino, M. Palesi, D. Patti, Noxim: Network-on-Chip Simulator, ⟨http://noxim.sourceforge.net/⟩.
[14] L. Zhe, L. Xiang, NoC mapping based on chaos artificial bee colony optimization, Proc. Comput. Probl. Solving (2011) 518–521.
[15] F. Khalili, H.R. Zarandi, A fault tolerant low energy multi application mapping onto NoC based multiprocessors, Proc. Comput. Sci. Eng. (2012) 421–428.
[16] J. Wang, L. Li, H. Pan, et al., Latency aware mapping for 3D NoC using rank based multi objective genetic algorithm, Proc. ASIC (2011) 413–416.
[17] L. Moller, L.S. Indrusiak. Comparative analysis of dynamic task mapping heuristics in heterogeneous NoC based MPSoCs. International Symposium on System on Chip, 2012, pp: 1-4.
[18] L. Tang, S. Kumar, A two-step genetic algorithm for mapping task graphs to a network on chip architecture, Proc. ESDSD (2003) 180–187.
[19] M. Mandelli, L. Ost, G. Sassatelli. Trading off system load and communication in mapping heuristics for improving NoC based MPSoCs reliability. Proceedings of the Sixteenth International Symposium on Quality Electronic Design, 2015, pp: 392-396.
[20] D. Zhu, L. Chen, T.M. Pinkston, et al., TAPP: temperature aware application mapping for NoC based manycore processors[C], Proc. Des. Autom. Test. Eur. (2015) 1241–1244.
[21] Y. Chen, J. Hu, X. Ling, Topology and mapping codesign for complex communication systems on wireless NoC platforms, Proc. Ind. Electron. Appl. (2013) 1442–1447.