Abstract: Optical network-on-chip is considered to be a promising technology to solve the problems of low bandwidth and high latency in the traditional interconnection network. However, due to the inevitable leakage of optical devices, the optical signal will receive crosstalk noise during transmission. In this paper, a heuristic fusion mapping algorithm PSO_SA for crosstalk optimization is proposed. First, the initial optimal mapping is obtained by particle swarm optimization, and then the local optimization of the mapping scheme is removed by combining with simulated annealing algorithm. The experimental results show that the crosstalk optimization performance of PSO_SA algorithm is better than that of GA algorithm in 263 dec, Wavelet, DVOPD and other applications, and the maximum optimization degree is 28.7%.

Keywords: ONoC; crosstalk optimization; mapping algorithm; heuristic optimization algorithm

1. Introduction

The improvement in the degree of chip crowding promotes the generation of network on chip [1,2]. However, as the number of cores grows, the common electrical network-on-chip cannot meet the requirements of advanced systems due to their low bandwidth and high power consumption [3]. Instead of the traditional network-on-chip (NoC) based on electrical interconnects, the optical NoC (ONoC) [4–8] provides higher communication bandwidth, as well as lower transmission delay and power costs.

Optical NoC has many advantages, but it also has some disadvantages that cannot be ignored. Crosstalk noise is the inherent characteristic of photonic devices [9,10]. Although the crosstalk noise is very small [11,12], it is a factor that cannot be ignored in the highly integrated on-chip system. Crosstalk noise is critical to the design of a photonic NoC architecture, as high crosstalk noise may easily result in an inoperable architecture [7,13,14]. According to [15], application-specific mapping optimization provides an important means for facing the crosstalk problem in application-specific multicore systems-on-chip (SoCs). [16] summarizes the task mapping strategy of the network on chip, including particle swarm optimization (PSO) [17–20], genetic algorithm (GA) [21], ant colony algorithm (ACO) [22], binomial mapping strategy (BMAP) [23], chain mapping strategy (CHMAP) [24] and so on. In [25], a mapping algorithm based on genetic algorithm is proposed to optimize the crosstalk performance of optical network-on-chip.

The purpose of designing an optimization mapping algorithm is to map each task in the application program to each node in the topology, and to find the optimal network performance. In the application, the communication between tasks can be transformed into a communication graph \( CG = G(C,E) \), which is a directed graph, in which each vertex \( c_i \in C \) is a task, \( e_{ij} \in E \) is the edge between task \( c_i \) and \( c_j \), representing the communication between them. Similarly, the NoC architecture of optical
interconnection can be represented by the connectivity topological graph \( X(T, L) \) described by the topology, routing algorithm and optical router mechanism, which represents the connection mode between the topological nodes, where \( t_i \in T \) represents the topological nodes in the network on the optical chip, and \( l_{ij} \in L \) is the physical link connecting the topological nodes \( t_i \) and \( t_j \). The mapping from communication graph \( CG = G(C,E) \) to topology graph \( X(T, L) \) can be defined by one-to-one mapping function on the premise of \( |C| \leq |T| \):

\[
\Omega: C \rightarrow T, \text{ s.t. } \Omega(c_i) = t_i, \forall c_i \in C, \exists t_i \in T
\]  

(1)

This definition means that each task in an application should be mapped to a topology node, and each topology node only corresponds to one task. Figure 1 shows the mapping of an application with eight tasks in mesh topology.

![Figure 1. Application mapping diagram.](image1)

Because the task communication relationship in an application is fixed, when the task is mapped to the on-chip network by mapping algorithm, the relative position of tasks will be determined. The crosstalk in the on-chip network is largely from the communication intersection, and the relative position of tasks will greatly affect the communication intersection of tasks. Therefore, the crosstalk in the on-chip network can be reduced by optimizing the mapping algorithm. When the mapping algorithm is used to optimize the crosstalk in the optical network, the evaluation value of the mapping algorithm needs to correspond to the crosstalk performance of the network, so as to improve the evaluation value as the standard to find the optimal mapping method.

In this paper, we design a heuristic fusion mapping algorithm PSO_SA to optimize crosstalk. The optimal mapping scheme is determined by two steps. First, the initial optimal mapping is obtained by the particle swarm optimization algorithm, and then the local optimization of the mapping scheme is removed by combining the simulated annealing algorithm.

2. Design of Heuristic Fusion Mapping Algorithm for Crosstalk Optimization

The general idea of our mapping algorithm is to integrate particle swarm optimization and simulated annealing algorithm (SA) [26–28] and combine the expansibility of PSO with the avoidance of local optimality of SA. The purpose of the mapping algorithm is to map the application tasks to the topology nodes. The relationship between the tasks is represented by the communication graph, and the relationship between the topology nodes is represented by the topology graph [29]. Our mapping algorithm is mainly for two-dimensional topological networks. Firstly, the two-dimensional topological network of \( M \times N \) is transformed into the arrangement of straight lines. The topological nodes are numbered as 1, 2, 3, \ldots, \( M \times N - 1 \), \( M \times N \), as shown in Figure 2. The final mapping scheme needs to place the task in a grid in the graph.

![Figure 2. Two-dimensional topological line arrangement.](image2)
The mapping algorithm needs to constantly adjust the task position in the topology structure, through which the crosstalk of optical network-on-chip is gradually reduced. Therefore, the method to adjust the task location is very important in the mapping algorithm. The mapping algorithm we designed includes two parts—one is to use particle swarm optimization, the other is to use simulated annealing optimization.

In particle swarm optimization, each dimension of particle velocity and position corresponds to a task in an application. The update mode of particle speed and position is shown in Equations (2) and (3).

\[ v_i = \omega v_i + c_1 \times r_1 \times (p_{b_i} - x_i) + c_2 \times r_2 \times (g_{b} - x_i) \]  (2)

\[ x_i = x_i + v_i \]  (3)

where \( v_i \) and \( x_i \) are the velocity and position of particles, respectively; \( \omega \) is the inertia factor; \( c_1 \) and \( c_2 \) are individual and whole learning factors respectively, equal to 1; \( r_1 \) and \( r_2 \) are random numbers between 0 and 1; \( p_{b_i} \) is the current optimal solution of a particle; and \( g_{b} \) is the global optimal solution of particle swarm. The adjustment of \( \omega \) is in accordance with Equation (4).

\[ \omega = e^{-30 \times \left( \frac{\text{run}}{\text{run}_{\text{max}}} \right)^4}, s \in [1, 30] \]  (4)

where run is the current number of iterations; \( \text{run}_{\text{max}} \) is the total number of iterations; and \( s \) is equal to 15 when the algorithm is designed.

Each dimension of particle speed and position corresponds to a task in the application. When an application mapping is completed, each task corresponds to a labeled topology node. When the speed of the corresponding dimension of a task in an application is greater than 0, the mapping position of the task moves to the right end of the line topology. Otherwise, the mapping position of the task moves to the left end of the line topology, and moves a certain distance according to the absolute value of the speed. Two points need to be paid attention to during the implementation of the actual algorithm:

1. when the positions coincide, move to the right end and left end, respectively, according to the positive and negative speed until the idle topology node is found;
2. when the left and right sides are out of bounds, find the mapping position by modulus value.

Figure 3 shows the movement adjustment of a nine-task application map. Suppose that the initial task mapping relationship of the application is task 1 corresponding to topology node 1, task 2 corresponding to topology node 2, and so on. Task 9 corresponding to topology node 9, and the speeds of task 1 to task 9 are +2, +1, −3, +5, −1, −3, +2, +1, +4, respectively.
It can be seen from the figure that the speed value of task 1 is “+2”, so the position moves two grids to the right to reach topology node 3; the speed value of task 2 is “+1”, and moves one grid to the right, but the topology node has been occupied by task 1, so task 2 continues to move to the right until the empty topology node 4 is found. The speed value of task 3 is “−3”, which moves three spaces to the left, but causes the boundary crossing, so the modulus reaches the position of topology node 9. The rest of the tasks are moved according to this rule. After nine moves, they become the mapping situation corresponding to step 9 in the figure.

After the particle swarm optimization algorithm generates the optimal solution, simulated annealing is needed to prevent it from falling into the local minimum. In the implementation, two tasks in the random exchange mapping solution are used to avoid the local optimal solution, which is similar to the gene mutation mode in the genetic algorithm. Figure 4 shows this process. On the basis of the optimal particle mapping solution, task 5 and task 9 are exchanged, and the network crosstalk value is calculated for the adjusted mapping solution.

The probability of modifying the deteriorating solution, the local optimal can be accepted with a certain probability. By introducing the deteriorating solution, the local optimal can be effectively avoided.

Considering that the simulated annealing algorithm can make up for the particle swarm optimization algorithm, the two algorithms can be combined in the design of network mapping algorithm on chip. The operation flow of our heuristic fusion mapping algorithm (Algorithm 1) is shown in Figure 5, and can be summarized into nine steps.

Step 1: Generate some mapping schemes randomly.
Step 2: Initialize the minimum crosstalk and optimal mapping of each particle. Compare to get the global and historical minimum crosstalk and optimal mapping. Mark the optimal particle.

Step 3: Choose different optimization methods according to whether it is the optimal particle. If it is the optimal particle, skip to step 6, otherwise step 4.

Step 4: Update the particle’s speed and position with reference to the above rules.

Figure 5. Crosstalk optimization process.
Step 5: Calculate crosstalk noise of each particle, update the minimum crosstalk noise of each particle, record the mapping mode at the same time, skip to step 8.

Step 6: Randomly exchange the mapping positions of two tasks in the optimal particle. If the crosstalk at this time is less than the minimum crosstalk of the particle, update the minimum crosstalk and mapping of the particle, otherwise update with a certain probability.

Step 7: Determine whether all particles in the particle swarm have been updated. If yes, perform step 9. Otherwise, go to step 3.

Step 8: Update the global minimum crosstalk noise and mapping mode, and relabel the optimal particles. If the global minimum crosstalk is less than the historical minimum crosstalk, update the historical minimum crosstalk and historical optimal mapping mode.

Step 9: At the end of the iteration, the optimal solution is returned. Otherwise, the inertia factor $\omega$ of PSO and the temperature parameters $t$ of SA are updated according to the above rules and the third step is executed.

Below is the pseudo code:

**Algorithm 1.**

**Input:** Iteration number: $I_T$; Population size: $N_P$

**Parameter:** Inertia factor: $\omega$; Learning factor: $C_1$, $C_2$; Random value: $R_1$, $R_2$; Temperature control value: $T$; Individual minimum crosstalk: $C_P$; Global minimum crosstalk: $C_G$; Current iteration number: $I_C$; Number of traversed particles: $N_C$; Individual optimal mapping: $M_P$; Global optimal mapping: $M_G$; Current particle label: $L_C$; Optimal particle label: $L_O$

**Output:** Historical minimum crosstalk: $C_H$; Historical optimal mapping: $M_H$

**Procedure**

- Generate mapping solutions of initial particle swarm randomly;
- Calculate the crosstalk corresponding to each particle mapping;
- Record the initial minimum crosstalk $C_P$ and mapping mode $M_P$;
- Obtain global minimum crosstalk $C_G$ and global optimal mapping $M_G$ by comparison and label the optimal particle $L_O$;
- Initialize historical minimum crosstalk $C_H$ and historical optimal mapping $M_H$;

**while** ($I_C < I_T$)

**while** ($N_C < N_P$)

**if** ($L_C == L_O$)

Exchange two task mapping positions randomly and calculate the crosstalk;

**if** (crosstalk < $C_P$)

Update $C_P$ and $M_P$;

**else**

Update $C_P$ and $M_P$ according to the Metropolis criterion

endif

else

Update speed and position of each particle;

Calculate the crosstalk of particle and update $C_P$, $M_P$

endif

end

Update $C_G$ and $M_G$ by comparison;

Update $C_H$ and $M_H$;

Change $\omega$ and $T$

end

**return** $M_H$

**4. Simulation Results**

We used the simulation software PhoNocMap [30] to build the network crosstalk simulation platform. In different applications [29] and topologies, the designed mapping algorithm was compared
with particle swarm optimization mapping algorithm and genetic mapping algorithm. The Optimized Unfolded Torus topology used in the experiment was mentioned in [31]. The parameter configuration of the simulation environment is shown in Table 1. Some multimedia applications will be used in the experiment, and their structure is shown in Figure 6.

| Table 1. Simulation parameters. |
|-------------------------------|----------------|
| Iteration Times               | 1000           |
| Population Size               | 100            |
| Routing Mode                  | XY order routing |
| Chip Size                     | 400 mm²        |
| Modulation Rate               | 10 Gb/s        |
| Laser Source Efficiency       | 30%            |
| Photodetector Sensitivity     | -14.2 dBm      |

![Figure 6. Multimedia application structures.](image)

The applications used in the experiment and their topological scale were referred to in [29,32]. The topology scale of PIP is $3 \times 3$, 263 dec, 263 enc, MPEG4, MWD and VOPD are $4 \times 4$, the topology scale of Wavelet is $5 \times 5$, and the topology scale of DVOPD is $6 \times 6$. The experimental results are shown in Figure 7.
The performance of the PSO algorithm is better than that of the PSO algorithm and GA algorithm in the optimization of crosstalk in different topologies and different applications. Compared with the GA algorithm, in addition to the application of MPEG4 and pip, wavelet and DVOPD, the crosstalk is reduced by 7.3%, 4.8%, 3.0%, 10.6% and 28.7%, respectively, so PSO_SA is better than the PSO algorithm and GA algorithm in the optimization of crosstalk in different applications.

4.1. Topology

Figure 7a shows the crosstalk comparison of mapping algorithms under different topologies, and the application uses 263 dec. It can be seen that in the comparison of different topological structures, the crosstalk optimization performance of PSO algorithm and GA algorithm is not stable. In the mesh and optimized unfolded torus structure, the performance of PSO algorithm is better than that of GA algorithm, while in the folded torus and unfolded torus structure, the performance of PSO algorithm is lower than that of GA algorithm. However, the crosstalk optimization performance of PSO algorithm is stable, which is better than the PSO algorithm and GA algorithm. Compared with the PSO algorithm and GA algorithm, the crosstalk optimization performance of the PSO algorithm is 7.2%, 1.5%, 0.4% and 2.3% lower than mesh, folded torus, unfolded torus and optimized unfolded torus respectively, so the performance of the PSO algorithm is better than that of the PSO algorithm and GA algorithm.

4.2. Multimedia Application

Figure 7b shows the crosstalk comparison of mapping algorithms under different applications. The topology is mesh. It can be seen that in the comparison of different applications, the overall crosstalk optimization performance of the PSO algorithm is the worst; the main reason for this is that the PSO algorithm is easy to fall into local minimum, and there is no obvious local minimum problem in the GA algorithm, so the overall performance of crosstalk optimization is better than the PSO algorithm. Based on the PSO algorithm, PSO_SA improves the local minimum problem in the PSO algorithm, and is better than the PSO algorithm and GA algorithm in the performance of crosstalk optimization. Compared with the GA algorithm, in addition to the application of MPEG4 and pip, MWD, and 263 dec, 263 enc, VOPD, Wavelet and DVOPD, the crosstalk is reduced by 7.3%, 4.8%, 3.0%, 10.6% and 28.7%, respectively, so PSO_SA is better than the PSO algorithm and GA algorithm in the optimization of crosstalk in different applications.
5. Conclusions

In this paper, a heuristic fusion mapping algorithm for crosstalk optimization is proposed, which combines particle swarm optimization (PSO) and simulated annealing (SA) to obtain a global optimal solution that is not easy to fall into local minimum, so as to reduce the crosstalk of the optical network-on-chip. Experimental results show that the crosstalk optimization performance of PSO-SA mapping algorithm is better than that of PSO mapping algorithm and GA mapping algorithm, both in topology comparison and application comparison. Therefore, the validity of the mapping algorithm is proved.

Author Contributions: Conceive and structure of the concept of this paper, X.S.; Resources, F.G. and N.W.; Supervision, N.W.; Writing—Original draft, X.S.; Writing—Review and editing, M.R.Y. and F.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Fundamental Research Funds for Central Universities (No. NP2019102).

Acknowledgments: The authors would like to thank Mahammad Rehan Yahya for his beneficial suggestions and comments.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Kogel, T.; Doerper, M.; Wieferink, A.; Leupers, R.; Ascheid, G.; Meyr, H.; Goossens, S. A modular simulation framework for architectural exploration of on-chip interconnection networks. In Proceedings of the 1st IEEE/ACM/IFIP International Conference on Hardware/Software Codesign and System Synthesis, Newport Beach, CA, USA, 1–3 October 2003; pp. 7–12.
2. Benini, L.; De Micheli, G. Networks on chips: A new SoC paradigm. *Computer* 2002, 35, 70–78. [CrossRef]
3. O’Connor, I. Optical solutions for system-level interconnect. In Proceedings of the 2004 International Workshop on System Level Interconnect Prediction, Paris, France, 14–15 February 2004. [CrossRef]
4. Shacham, A.; Bergman, K.; Carloni, L.P. Photonic networks-on-chip for future generations of chip multiprocessors. *IEEE Trans. Comput.* 2005, 57, 1246–1260. [CrossRef]
5. Sherwood-Droz, N.; Wang, H.; Chen, L.; Lee, B.G.; Biberman, A.; Bergman, K.; Lipson, M. Optical 4 × 4 hitless silicon router for optical networks-on-chip (NoC). *Opt. Express* 2008, 16, 15915–15922. [CrossRef] [PubMed]
6. Xie, Y.Y.; Zhao, W.L.; Xu, W.H.; Huang, Y.X.; Zhang, Z.D. Performance optimization and evaluation for mesh-based optical networks-on-chip. *IEEE Photonics J.* 2015, 7, 1–12. [CrossRef]
7. Xie, Y.; Nikdast, M.; Xu, J.; Wu, X.; Zhang, W.; Ye, Y.; Wang, X.; Wang, Z.; Liu, W. Formal worst-case analysis of crosstalk noise in mesh-based optical networks-on-chip. *IEEE Trans. Very Large Scale Integr. VLSI Syst.* 2012, 21, 1823–1836. [CrossRef]
8. Gu, H.; Chen, K.; Yang, Y.; Chen, Z.; Zhang, B. MRONoC: A low latency and energy efficient on chip optical interconnect architecture. *IEEE Photonics J.* 2017, 9, 1–12. [CrossRef]
9. Xia, F.; Rooks, M.; Sekaric, L.; Vlasov, Y. Ultra-compact high order ring resonator filters using submicron silicon photonic wires for on-chip optical interconnects. *Opt. Express* 2007, 15, 11934–11941. [CrossRef] [PubMed]
10. Bogaerts, W.; Dumon, P.; Van Thourhout, D.; Baets, R. Low-loss, low-cross-talk crossings for silicon-on-insulator nanophotonic waveguides. *Opt. Lett.* 2007, 32, 2801–2803. [CrossRef] [PubMed]
11. Chen, H.; Poon, A.W. Low-loss multimode-interference-based crossings for silicon wire waveguides. *IEEE Photonics Technol. Lett.* 2006, 18, 2260–2262. [CrossRef]
12. Bogaerts, W.; Dumon, P.; Van Thourhout, D.; Baets, R. Compact, low-loss waveguide crossings for high-index-contrast SOI photonic wires. In Proceedings of the Optical Fiber Communication Conference, Anaheim, CA, USA, 25–29 March 2007.
13. Xie, Y.; Nikdast, M.; Xu, J.; Zhang, W.; Li, Q.; Wu, X.; Ye, Y.; Wang, X.; Liu, W. Crosstalk noise and bit error rate analysis for optical network-on-chip. In Proceedings of the Design Automation Conference, Anaheim, CA, USA, 13–18 June 2010; pp. 657–660.
14. Ye, Y.; Xu, J.; Wu, X.; Zhang, W.; Liu, W.; Nikdast, M. A torus-based hierarchical optical-electronic network-on-chip for multiprocessor system-on-chip. ACM J. Emerg. Technol. Comput. Syst. 2012, 8, 5. [CrossRef]

15. Werner, S.; Navaridas, J.; Luin, M. Amon: An advanced mesh-like optical noc. In Proceedings of the 2015 IEEE 23rd Annual Symposium on High-Performance Interconnects, Santa Clara, CA, USA, 26–28 August 2015; pp. 52–59.

16. Sahu, P.K.; Chattopadhyay, S. A survey on application mapping strategies for network-on-chip design. J. Syst. Archit. 2013, 59, 60–76. [CrossRef]

17. Kennedy, J.; Eberhart, R. Particle swarm optimization (PSO). In Proceedings of the IEEE International Conference on Neural Networks, Perth, Australia, 27 November–1 December 1995; pp. 1942–1948.

18. Eberhart, R.C.; Shi, Y. Comparison between genetic algorithms and particle swarm optimization. In Proceedings of the International Conference on Evolutionary Programming, San Diego, CA, USA, 25–27 March 1998; pp. 611–616.

19. Shi, Y.; Eberhart, R.C. Parameter selection in particle swarm optimization. In Proceedings of the International Conference on Evolutionary Programming, San Diego, CA, USA, 25–27 March 1998; pp. 591–600.

20. Eberhart, R.C.; Shi, Y. Comparing inertia weights and constriction factors in particle swarm optimization. In Proceedings of the 2000 Congress on Evolutionary Computation, La Jolla, CA, USA, 16–19 July 2000; pp. 84–88.

21. Lei, T.; Kumar, S. A two-step genetic algorithm for mapping task graphs to a network on chip architecture. In Proceedings of the Euromicro Symposium on Digital System Design, Belek-Antalya, Turkey, 1–6 September 2003; pp. 180–187.

22. Wang, J.; Li, Y.; Chai, S.; Peng, Q. Bandwidth-aware application mapping for NoC-based MPSoCs. J. Comput. Inf. Syst. 2011, 7, 152–159.

23. Shen, W.T.; Chao, C.H.; Lien, Y.K.; Wu, A.Y. A new binomial mapping and optimization algorithm for reduced-complexity mesh-based on-chip network. In Proceedings of the First International Symposium on Networks-on-Chip (NOCS’07), Princeton, NJ, USA, 7–9 May 2007; pp. 317–322.

24. Tavanpour, M.; Khademzadeh, A.; Janidarmian, M. Chain-Mapping for mesh based Network-on-Chip architecture. IEICE Electron. Express 2009, 6, 1535–1541. [CrossRef]

25. Fusella, E.; Cilardo, A. Crosstalk-aware mapping for tile-based optical network-on-chip. In Proceedings of the 2015 IEEE 17th International Conference on High Performance Computing and Communications, 2015 IEEE 7th International Symposium on Cyberspace Safety and Security, and 2015 IEEE 12th International Conference on Embedded Software and Systems, New York, NY, USA, 24–26 August 2015; pp. 1139–1142.

26. Aarts, E.; Korst, J. Simulated Annealing and Boltzmann Machines; Wiley: New York, NY, USA, 1988.

27. Johnson, D.S.; Aragon, C.R.; McGeoch, L.A.; Schevon, C. Optimization by simulated annealing: An experimental evaluation; part I, graph partitioning. Oper. Res. 1989, 37, 865–892. [CrossRef]

28. Hwang, C.R. Simulated annealing: Theory and applications. Acta Appl. Math. 1988, 12, 108–111.

29. Fusella, E.; Cilardo, A. Crosstalk-aware automated mapping for optical networks-on-chip. ACM Trans. Embed. Comput. Syst. 2016, 16, 16. [CrossRef]

30. Fusella, E.; Cilardo, A. PhoNoCMap: An application mapping tool for photonic networks-on-chip. In Proceedings of the 2016 Conference on Design, Automation & Test in Europe, Dresden, Germany, 14–18 March 2016; pp. 289–292.

31. Feng, K.; Ye, Y.; Xu, J. A formal study on topology and floorplan characteristics of mesh and torus-based optical networks-on-chip. Microprocess. Microsyst. 2013, 37, 941–952. [CrossRef]

32. Fusella, E.; Cilardo, A. Minimizing power loss in optical networks-on-chip through application-specific mapping. Microprocess. Microsyst. 2016, 43, 4–13. [CrossRef]

© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).